Problem

Show that if P = NP, then P = PH.

Step-by-step solution

Step 1 of 1

It is required to show that if $P = NP,$ then $P = PH$.

• Firstly, if $P = NP$, then because P is closed under complement, thus $P = C_o NP$. Written as, $P = \Sigma_1 P = \Pi_1 P$.

• Now using induction that if $P = \Sigma_i P = \Pi_i P$, then $P = \Sigma_{i+1} P = \Pi_{i+1} P$.

1. Assume $\Sigma_{i+1} P$ machine M ,that consists of a run of the existential branching, then existential branching etc.

2. Assume the computation sub-tree path whose root are first universal step along path. For each such type of sub-tree, M is performing a Π_i computation. By hypothesis, $\Pi_i P = P$.

3. Thus for the forming of new machine S each of computation sub-trees can be replaced by deterministic (non- branching) polynomial time of computation.

4. If assume $a(n)$ be the number of maximum steps which are taken by other machine before the start of universal machine, $P(n)$ be the maximum steps which are taken by any deterministic which were substituted for Π_i computations in P machines, therefore S is covered by $a(n) + p(a(n))$. Remember that the $p(a(n))$ term is composition of the functions, because P sub procedures with inputs are computing which may be a longer than n (but it must be equal or smaller than $a(n)$, since only $a(n)$ steps are executed on the time the sub procedures are used).

5. Since a and P both are polynomials, Therefore, S is in NP. By hypothesis $P = NP$, so S is in P as well.

6. A similar type of argument may be used to reduce a $\Pi_{i+1} P$ machine to $P = C_o NP$ machine , hence, putting it in P as well, and completing collapse of hierarchy.

Comment