Problem

Show how to compute the descriptive complexity of strings K(x) with an oracle for A_{TM}.

Step-by-step solution

Step 1 of 2

Descriptive complexity of strings:-

If *x* be binary string, then the minimal description and descriptive complexity of *x*'s are $d(x)$ and $K(x)$ respectively. Turing machine *M* and small string *w* we get minimal description is $\langle M, w\rangle$. From several of such shorter strings we select lexicographically among them then we can get descriptive complexity of such strings $K(x) = |d(x)|$.

Comment

Step 2 of 2

Now we have to show how to compute $K(x)$ with an oracle for A_{TM}.

• For the given string *x*, start testing all the strings '*S*' up to the length $|x|+c$

Where $c =$ length of *TM* (Turing machine) that halts immediately upon starting.

• All the strings up to the length $|x|+c$ are potential description of *x*.

• If *S* is well formed as $\langle M, w\rangle$ from all binary strings in lexicographic order, then we simulate *M* with input *w* and see if it halts with *x* on the tape.

• Here we do not know whether *M* will halt on input *w* or not.

• An oracle for A_{TM} can determine this.

$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a } TM \text{ and } M \text{ accepts } w\}$

• An oracle for A_{TM} will take $\langle M, w\rangle$ as input and determine whether *M* accepts *w* or not.

• If *M* doesn't halt we move on to the next string *S*, and so on.

• After that we will find lexicographically first string *S* among them.

• In this way shortest string will be determined and it is represented as minimal description $d(x)$.

• From $d(x)$, we find $K(x)$ as

$K(x) = |d(x)|$

• By this procedure, we will compute $K(x)$ with an oracle for A_{TM}.

Comment