random Turing machine

Canonical name	RandomTuringMachine
Date of creation	2013-03-22 13:01:53
Last modified on	2013-03-22 13:01:53
Owner	Henry (455)
Last modified by	Henry (455)
Numerical id	5
Author	Henry (455)
Entry type	Definition
Classification	msc 68Q10
Defines	one sided error
Defines	two sided error
Defines	one-sided error
Defines	two-sided error
Defines	minimal error

A random Turing machine is defined the same way as a non-deterministic Turing machine, but with different rules governing when it accepts or rejects. Whenever there are multiple legal moves, instead of always guessing right, a random machine selects one of the possible moves at random.

As with non-deterministic machines, this can also be viewed as a deterministic machine with an extra input, which corresponds to the random selections.

There are several different ways of defining what it means for a random Turing machine to accept or reject an input. Let $\mathrm{Prob}_T(x)$ be the probability that T halts in an accepting state when the input is x.

A positive *one-sided error* machine is said to accept x if $\mathrm{Prob}_T(x) \geq \frac{1}{2}$ and to reject x if $\mathrm{Prob}_T(x) = 0$. A negative one-sided error machine accepts x if $\mathrm{Prob}_T(x) = 1$ and rejects x if $\mathrm{Prob}_T(x) \leq \frac{1}{2}$. So a single run of a positive one-sided error machine never misleadingly accepts but may misleadingly reject, while a single run of a negative one-sided error machine never misleadingly rejects.

The definition of a positive one-sided error machine is stricter than the definition of a non-deterministic machine, since a non-deterministic machine rejects when there is no certificate and accepts when there is at least one, while a positive one-sided error machine requires that half of all possible guess inputs be certificates.

A *two-sided error* machine accepts x if $\mathrm{Prob}_T(x) \geq \frac{2}{3}$ and rejects x if $\mathrm{Prob}_T(x) \leq \frac{1}{3}$.

The constants in any of the definitions above can be adjusted, although this will affect the time and space complexity classes.

A *minimal error* machine accepts x if $\mathrm{Prob}_T(x) > \frac{1}{2}$ and rejects x if $\mathrm{Prob}_T(x) < \frac{1}{2}$.

One additional variant defines, in addition to accepting states, rejecting states. Such a machine is called *zero error* if on at least half of all inputs it halts in either an accepting or a rejecting state. It accepts x if there is any sequence of guesses which causes it to end in an accepting state, and rejects if there is any sequence of guesses which causes it to end in a rejecting state. In other words, such a machine is never wrong when it provides an answer, but does not produce a decisive answer on all input. The machine can emulate a positive (resp. negative) one-sided error machine by accepting (resp. rejecting) when the result is indecisive.

It is a testament to the robustness of the definition of the Turing machine (and the Church-Turing thesis) that each of these definitions computes the same functions as a standard Turing machine. The point of defining all these types of machines is that some are more efficient than others, and therefore they define different time and space complexity classes.