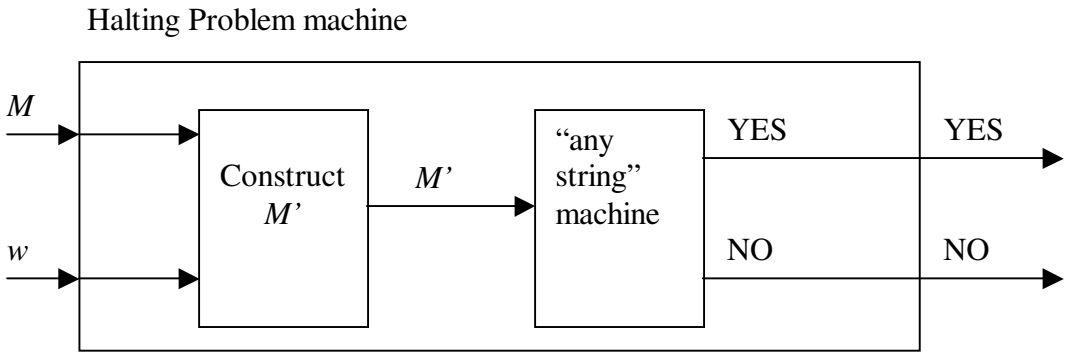
CSCI-2400 Models of Computation

Solutions to Homework 7

(Part of this solution was provided by Andrew Shilliday)

Problems

1. Prove that the problem of determining whether for a Turing machine M there is some input string for which M halts is undecidable.



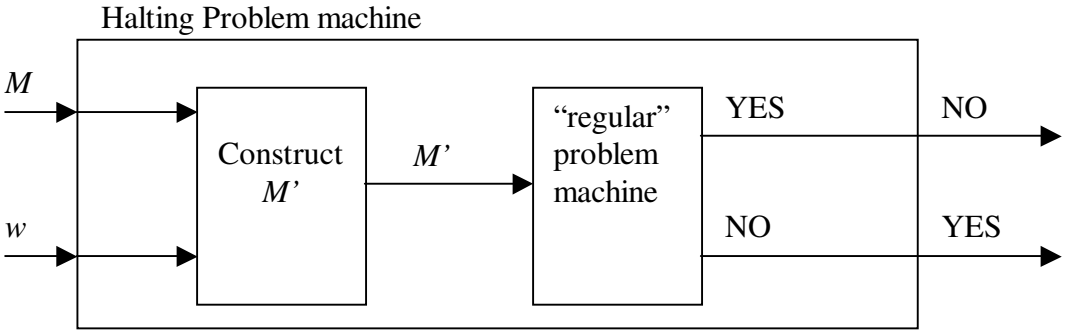
Let's call the problem for which we want to prove it is undecidable as the "any string" problem. We will reduce the halting problem to the "any string" problem, as shown in the above figure.

First, we take *M* and *w* and we construct a new machine *M'*, such that *M'* first simulates *M* on input *w*. If during the simulation *M* halts on input w, then *M'* enters a new set of states in which *M'* halts for some input string on the tape of *M'*. However, if during the simulation *M* doesn't halt on input *w,* then *M'* is stuck in the simulation of *M*, and thus *M'* will not halt either, for any input string on the tape of *M'*.

We now have constructed *M'* in such a way that if *M* halts on *w*, *M'* will halt on some string. Otherwise, if *M* doesn't halt on *w*, then *M'* will not halt as well. Therefore, if we now the answer to the question of whether *M'* halts on any input string, then we know the answer to the question of whether *M* halts on *w* or not.

Because we have reduced the halting problem to the "any string" problem, we can conclude that because the halting problem is undecidable, neither is the "any string" problem.

2. Prove that the problem of determining whether for a Turing machine M the language L(M) is regular is undecidable.



Let's call the problem which we want to prove is undecidable as the "regular" problem. We will reduce the halting problem to the "regular" problem, as shown in the above figure.

First, we take M and w and we construct a new machine *M'*, such that *M'* first simulates *M* on input *w*. If during the simulation *M* halts on *w*, then *M'* enters a new set of states in which M' accepts a non-regular language, like $a^n b^n$, on the tape of *M'*. Namely, in this case $L(M') = \{a^n b^n\}$ which is non-regular . Otherwise, if during the simulation *M* does not halt on *w*, then *M'* is stuck in the simulation of *M*, and thus *M'* will not halt either. In this case *M'* will not accept any string and thus $L(M') = \varnothing$, which is a regular language since there is a finite automaton with one state which doesn't accept any string.

We now have constructed *M'* in such a way that if *M* halts on *w*, then *L(M')* is a non-regular language. Otherwise, if *M* doesn't halt on *w*, then *L(M') is* a regular language. Therefore, if we know the answer to the question of whether *L(M')* is regular or not, then we know the answer to the question of whether *M* halts on *w* or not.

Therefore, the halting problem can be reduced to the "regular" problem, and because the halting problem is undecidable, so must be the "regular" problem.