$\textbf{HALT}_{\text{TM}}$

Prove that

$\text{HALT}_{\text{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ halts on input } w\}$

is undecidable.

Proof by reduction: A_{TM} to HALT_{TM}

Initial Assumption: There exists a TM, $\text{D}_{\text{HALT}}\langle M\rangle$, that can decide HALT_{TM}.

We will now build a decider for A_{TM} that takes input $\langle M, w\rangle$ and uses D_{HALT} to determine if M accepts w. This decider, D_{A}, will function as follows:

1.) Run D_{HALT} on input $\langle M, w\rangle$
2.) If D_{HALT} rejects, REJECT
3.) If D_{HALT} accepts, then simulate M on input w
  - If M accepts w, ACCEPT
  - If M rejects w, REJECT

Case 1: $\langle M, w\rangle \in A_{TM}$ In this case M accepts w and thus halts, so D_{HALT} will accept $\langle M, w\rangle$ and D_{A} will accept during the simulation of M on w.

Case 2: $\langle M, w\rangle \notin A_{TM}$ This means that M either rejects or loops on input w. If M loops on w then D_{HALT} will reject $\langle M, w\rangle$ causing D_{A} to reject. If M rejects w then D_{HALT} will accept $\langle M, w\rangle$ and D_{A} will reject during the simulation of M on w.

Given our initial assumption that HALT_{TM} was decidable, we were able to construct a decider for A_{TM}. However, we know that A_{TM} is undecidable. Thus we can conclude that our initial assumption was false and that HALT_{TM} is undecidable. ■