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: \( A_{\text{TM}} \) to \( \text{HALT}_{\text{TM}} \)

Initial Assumption: There exists a TM, \( D_{\text{HALT}} \langle M \rangle \), that can decide \( \text{HALT}_{\text{TM}} \).

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

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

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

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