Recall that a language L is _recognizable_ if there is a TM M for which $L(M) = L$. L is instead _co-recognizable_ if $L(M) = \overline{L}$. If L is both in RE (the recognizable languages) and in co-RE (the co-recognizable languages), then $L \in DEC$ (the decidable languages). If $L \notin DEC$, then at least one of $L \notin RE$ and $L \notin co\text{-}RE$ must be true.

We have seen that $A_{TM} \in RE$, $A_{TM} \notin DEC$ and $EQ_{TM} \in co\text{-}RE$ and $EQ_{TM} \notin DEC$. There are two more useful undecidable languages to look at before moving on to Rice's Theorem, a general statement about decidability.

$$HALT_{TM} = \{\langle M, w \rangle \mid M \text{ halts on } w\}$$

This is the halting problem. Like A_{TM}, it is recognizable. This is easy to show.

H = "On input $\langle M, w \rangle$, ← reject if improperly formatted
    1) Run M(w)
    2) Accept"

Clearly, $L(H) = HALT_{TM}$.

To see that $HALT_{TM}$ is not decidable, assume it is. Then there is a decider D for $HALT_{TM}$, that is $L(D) = HALT_{TM}$ and D halts on every input.

Using D, we can construct a decider A for A_{TM}. This is, of course nonsense, so no such D can exist. Now let's write down A.

A = " On input $\langle M, w \rangle$,

    1) Run $D(M, w)$

    2) If $D(M, w)$ rejects, reject

    3) Run $M(w)$

    4) If $M(w)$ accepts, accept

    5) Reject "

Notice that A is garunteed to halt, since if $M(w)$ doesn't halt, we skip steps 3-5. Then we have $L(A) = A_{TM}$, so A is a decider for A_{TM}.

$\rightarrow\leftarrow$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $\square$

The other language of interest to look at is

$$E_{TM} = \{ \langle M \rangle \mid L(M) = \emptyset \}.$$

$E_{TM} \in co\text{-}RE$ is easy to show, just like $EQ_{TM} \in co\text{-}RE$. We give a TM N for which $L(M) = \overline{E_{TM}}$ below.

  N = " On input $\langle M \rangle$,

    1) For $i = 1$ to ∞

      a) For $j = 1$ to i

        i) Run $M(w_j)$ for i steps

        ii) If $M(w_j)$ accepted, accept "

Eventually, if M accepts a string, N will find it and accept, ie $L(M) \neq \emptyset$.

We now show that E_{TM} is not decidable by reducing an undecidable language to it. as usual. Let D be a decider such that $L(D) = E_{TM}$.

Define the utility TM $R_{M,\nu}$ to be

$R_{M,\nu}$ = "On input w,

    1) If $w \neq \nu$, reject

    2) Run $M(\nu)$ and accept if M does"

Notice that $L(R_{M,\nu}) = \begin{cases} \varnothing & M(\nu) \text{ does not accept} \\ \{\nu\} & M(\nu) \text{ accepts.} \end{cases}$

We can now give a decider D' for A_{TM}.

D' = "On input $\langle M, \nu \rangle$,

    1) Run $D(R_{M,\nu})$

    2) If $D(R_{M,\nu})$ accepts, reject

    3) Accept "

$D(R_{M,\nu})$ accepts iff $M(\nu)$ does not accept. As such,

    $L(D') = \{\langle M, \nu \rangle \mid M(\nu) \text{ accepts}\} = A_{TM}.$

Moreover, every step of D' is guaranteed to complete, so D' decides A_{TM}.

$\rightarrow\leftarrow$

No such D exists, hence $E_{TM} \notin DEC.$

$\square$

Notice that this proof reduces A_{TM} not to E_{TM} but rather $\overline{E_{TM}}$. (We query whether or not $R_{M,\nu}$ is <u>not</u> empty). In fact, there is no "proper" reduction from A_{TM} to E_{TM} since $A_{TM} \in RE$ and $E_{TM} \notin RE$. What we mean by "proper" will come later, but keep this in mind. We reduce problems in RE to problems in RE ($\overline{E_{TM}} \in RE$ b/c $E_{TM} \in$ co-RE). Similarly, we reduce problems in co-RE to problems in co-RE.

Now you may have noticed that A_{TM}, EQ_{TM}, E_{TM}, and $HALT_{TM}$ are all questions about languages. This is no coincidence! Let's turn to Rice's Theorem.