Theorem: $HALT_{TM}$ is undecidable, but recognizable.

(Compare: A_{TM} " " , " ")

Today + other reductions

Wed, Fri – Old exam problems ("Office Hours")

→ {By topic
{Any problems, any day ← mostly

(Plus office hours on Monday, Dec 4.)

~~Decidable~~:

A <u>decider</u> is a TM that halts on each input.

<u>Halt</u> means eventually arrives at q_{acc} or q_{rej}

"<u>Not halting</u>" " never arriving " " " " ("<u>looping</u>")

Why is A_{TM} recognizable?

description of M, Turing machine

A_{TM} "acceptance problem" = $\{\langle M, w\rangle \mid$ M accepts w, i.e. on input w, M eventually reaches $q_{acc}\}$

standardized: $Q = \{1, \ldots, a\}, \Sigma = \{1, \ldots, b\}, \Gamma = \{1, \ldots, c\}$

$HALT_{TM}$ = "halting problem" = $\{\langle M, w\rangle \mid$ M halts on input $w\}$

$\overline{A_{TM}} = \Sigma^* \setminus A_{TM}$ is unrecognizable.

$\overline{HALT_{TM}}$ " "

Why is A_{TM} recognizable?

A Turing machine, M, recognizes the inputs on which it halts and reaches q_{acc}. [Otherwise it would halt and reach q_{rej} or not halt]

Given: $\langle M, w \rangle$ what kind of algorithm?

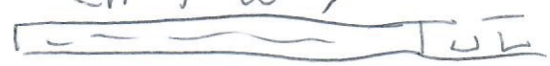
- must halt in q_{acc} if M accepts w
- either halts in q_{rej} or doesn't halt if M does not accept w. ?

Test: Description of M: Q: states $\{1, \ldots, a\}$
Σ: $\{1, \ldots, b\}$ δ - values
Γ: $\{1, \ldots, c\}$

Description of w

step 1 { 0

<M, w>



if M accepts w, you must halt, say "yes"
" " doesn't, you can say "no" or never halt

Algorithm uses "Universal TM"; run a UTM: simulate what M would do, step by step on w, then

1. if M reaches q_{acc} in M, the Universal TM halts and accepts
2. " " " q_{rej} " " , " " " halts and rejects
3. " " never halts, then the UTM never halts.

Using a Univ. TM to analyze what happens when given

$\langle M, w \rangle$ if M on input w $\begin{cases} \text{accepts} \\ \text{reject} \\ \text{loops (doesn't halt)} \end{cases}$

gives an algorithm to recognize A_{TM}, but this algorithm doesn't always halt. But UTM will not decide A_{TM}.

Similarly, a UTM can be used to recognize but not decide $HALT_{TM}$.

Remarks: ① $A_{TM} \leq_{\text{poly time}} HALT_{TM}$ ← How to do?

② $HALT_{TM} \leq_{\text{poly time}} A_{TM}$

Say you have a $\left\{\begin{matrix}\text{oracle} \\ \text{subroutine}\end{matrix}\right.$ to decide $HALT_{TM}$, could you decide A_{TM}?

$\langle M, w \rangle \in A_{TM}$? Does M, on input w, reach q_{acc}

$\begin{cases} \text{reaches } q_{acc} \\ \quad " \quad q_{rej} \\ \text{loops} \end{cases}$

Make M'

q_{acc}

introduce some infinite loop

don't have to do anything

M' on w halts $\Leftrightarrow$ M accepts w