

W9L3 decidability and reductions

Friday, April 3, 2020 9:14

So far we have A_{TM} and $HALT_{TM}$ which are both undecidable but recognizable.

In your book, they discuss:

- $E_{TM} = \{ \langle M \rangle \mid M \text{ is a Turing machine and } L(M) = \emptyset \}$
- $REGULAR_{TM} = \{ \langle M \rangle \mid M \text{ is a Turing machine and } L(M) \text{ is a regular language} \}$
- $EQ_{TM} = \{ \langle M_1, M_2 \rangle \mid M_1 \text{ and } M_2 \text{ are Turing machines and } L(M_1) = L(M_2) \}$
- $ALL_{CFG} = \{ \langle G \rangle \mid G \text{ is a CFG and } L(G) = \Sigma^* \}$

... and prove that all of these are not decidable.

Claim: E_{TM} is undecidable.

Proof: (by contradiction)

S'pose that E_{TM} is decidable, and is decided by some decider R .

Want to show how to build a decider for A_{TM} using R as a subroutine (this will be our contradiction).

Build $S =$ "On input $\langle M, w \rangle$ where M is a Turing machine and w is a string:

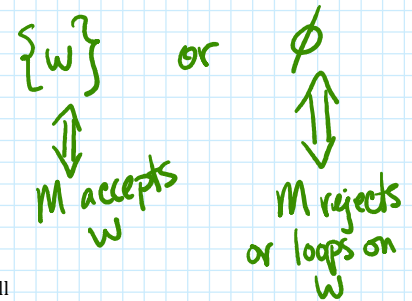
1. Build a new Turing machine N_1 .
 $N_1 =$ "on input x :
 - a. If $x \neq w$ then reject.
 - b. If $x = w$ then run M on input w and do the same."
2. Run R on input $\langle N_1 \rangle$.
3. If R accepts, reject. Else, accept."

Need to check:

- Is S a decider?
 Yes: it has 3 lines; line 1 definitely halts, line 2 halts because R is a decider, and line 3 halts and returns accept/reject, so overall S is a decider.
- If $\langle M, w \rangle \in A_{TM}$ then we know that M accepts string w . N_1 will reject all other strings, but accepts w , so $L(N_1) = \{w\}$. So on line 2, R will reject $\langle N_1 \rangle$, so on line 3, S accepts!
- If $\langle M, w \rangle$ is not in A_{TM} and M does not accept w . N_1 will accept no strings, so $L(N_1) = \emptyset$. So on line 2, R will accept $\langle N_1 \rangle$, so on line 3, S will reject.
- If $\langle M, w \rangle$ is not in A_{TM} because it is a bad encoding. Then N_1 does some weird behavior because M and w are a bad encoding, but it definitely doesn't accept ever, so $L(N_1) = \emptyset$, so again R will accept $\langle N_1 \rangle$ and so on line 3, S will reject.

So we have built a decider S for A_{TM} , but we know that A_{TM} is not decidable! $\Rightarrow \Leftarrow \square$

IDEA:
 $L(N_1)$ is either



We have R to distinguish between these 2 cases ($\{w\}$ or \emptyset) & use it to figure out the answer for the A_{TM} problem (M accepts w or doesn't).

Define $CONTEXTFREE_{TM} = \{ \langle M \rangle \mid M \text{ is a Turing machine and } L(M) \text{ is a context-free language} \}$

Claim: $CONTEXTFREE_{TM}$ is undecidable.

Proof: (by contradiction)

S'pose that $CONTEXTFREE_{TM}$ is decidable, and is decided by some decider R .

Want to show that we can build a decider for $HALT_{TM}$ using R as a subroutine (this will be our contradiction).

Build $S =$ "On input $\langle M, w \rangle$ where M is a Turing machine and w is a string:

1. Build a new Turing machine N_2 :
 $N_2 =$ "On input x :
 - a. If $x = 0^n 1^n 2^n$ for some n , accept.
 - b. Else, run M on input w .
 - c. Accept."
2. Run R on $\langle N_2 \rangle$ and do the same."

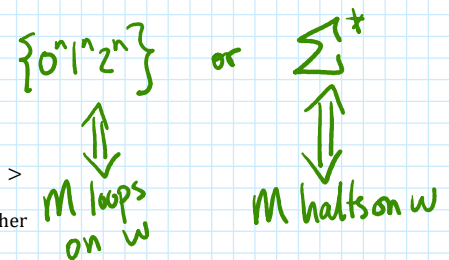
Need check:

- S is a decider. Line 1 finishes and line 2 runs a decider, so it also finishes.
- If $\langle M, w \rangle \in HALT_{TM}$ then M halts on w . So N_2 will accept Σ^* , which is context-free, so R will accept $\langle N_2 \rangle$ so S will also accept.
- If $\langle M, w \rangle$ is not in $HALT_{TM}$ and M loops on w , then N_2 will accept strings in $\{0^n 1^n 2^n\}$ and loop on all other strings, so $L(N_2)$ is not context-free. So R will reject $\langle N_2 \rangle$, so S will also reject.
- If $\langle M, w \rangle$ is not in $HALT_{TM}$ and is badly formatted, then we should specify either (I) "when you try to run something that's not a TM on line (b), you loop forever" or (II) "when you try to run something that's not a TM on line (b), you reject". Then N_2 accepts $\{0^n 1^n 2^n\}$ and loops/rejects all other strings, so again $L(N_2)$ is not context-free, so R will reject $\langle N_2 \rangle$ so S will also reject.

So S is a decider for $HALT_{TM}$, but $HALT_{TM}$ is undecidable! $\Rightarrow \Leftarrow \square$

IDEA:

$L(N_2)$ is either



We have R to distinguish between these ($\{0^n 1^n 2^n\}$ or Σ^*) & use it to figure out the answer

for HALT_{TM} (does M loop on w or not?).