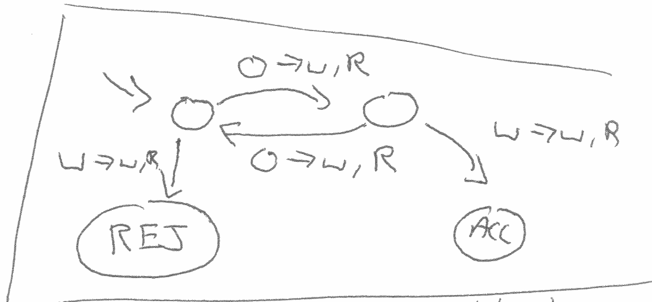
23-1

REG $\subset$ ~~TM~~ Σ_0

↓ a language that has a DFA

↓ a language that has a decider

DFA for odd strings



TM for add strings

compile a DFA into a TM decider

1. move right (write blank on whatever)
2. on blank, look at F and go to ACC / REJ

$\delta(q_i, \sqcup) = (ACC, \sqcup, R)$ if $q_i \in F$

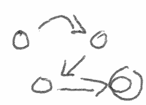
$\delta(q_i, \sqcup) = (REJ, \sqcup, R)$ if $q_i \notin F$

Cooler: 1 TM that is all REGs (interpreter)

$\langle \quad \rangle$ = a magical encoding function

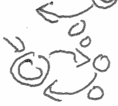
$\langle 5, 2 \rangle \rightarrow$ 00000#00

$\rightarrow$ ☺☹☺♡☺☹

$\langle$  $\rangle \rightarrow$ 0100011100

$A_{DFA} = \{ \langle D, w \rangle \mid D \in DFA, w \in D.\Sigma^*$, D accepts $w \}$

$\langle$  , 00 $\rangle \notin A_{DFA}$

$\langle$  , 00 $\rangle \in A_{DFA}$

A for <u>acceptance</u>

$\langle$  , 000 $\rangle \in A_{DFA}$

1. Decode input
2. Look at first char in input tape (ie w)
3. Consult δ and change the state tape
4. If the char is $\sqcup$, consult F

Tapes: D, w, q_i

↑ state tape

23-2

$A_{NFA} = \{ \langle N, w \rangle \mid N \in NFA,\ w \in \Sigma^*,\ N \text{ accepts } w \}$

Option 1: use non-det TM

Option 2: $\underbrace{\text{Compile } N \text{ to } D}_{\text{TM transducer}}$, then use A_{DFA}

$A_{REX} = \{ \langle R, w \rangle \mid R \in REX,\ w \in \Sigma^*,\ R \text{ accepts } w \}$

option: use compiler to NFA, then use A_{NFA}

$A_{DFA, NFA, REX}$ are all decidable ($\in \Sigma_0$)

$X \subset \Sigma_0$ (compiler)

$A_X \in \Sigma_0$ (interpreter exists, is correct, and a decider)

$E_{DFA} = \{ \langle D \rangle \mid D \in DFA \text{ and } L(D) = \emptyset \}$

Emptiness

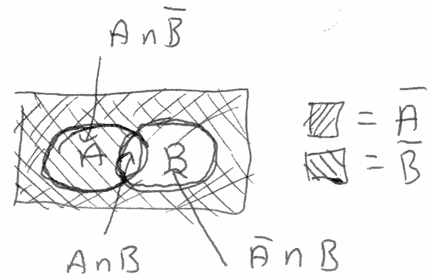
1. If F is empty
2. Look at stuff in F

3. Is there a path from q_0 to $q_f \in F$?
   Do a DFS on the graph (Q, δ) from q_0

$EQ_{DFA} = \{ \langle A, B \rangle \mid A \in DFA,\ B \in DFA,\ L(A) = L(B) \}$

$$\left(\begin{matrix} (A \cap \overline{B}) \cup \\ (\overline{A} \cap B) \end{matrix} \right) = C \qquad L(C) = \emptyset$$

$E_{DFA}(C)$



Model-Checking

A = Program / Model

B = Specification