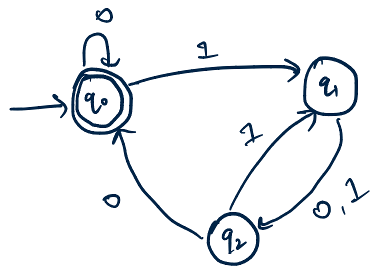
Discussion Notes Week 7 CSE105 Spring '23

Agenda: 4.1, 4.3, 4.4, 4.5, 4.30 from Sipser

4.1

let M =



Recall:

$A_{DFA} = \{\langle M, w\rangle \mid M$ is a DFA that accepts $w\}$

(a) Is $\langle M, 0100\rangle \in A_{DFA}$?

Yes. On input 0100, M ends in state q_0 which is an accepting state.

(b) Is $\langle M, 011\rangle \in A_{DFA}$?

No. On input 011, M ends in state q_2 which is not accepting.

(c) Is $\langle M\rangle \in A_{DFA}$?

No. Doesn't type check.

(d) Is $\langle M, 0100\rangle \in A_{REX}$? $A_{REX} = \{\langle R, w\rangle \mid R$ is a regular expression that generates $w\}$

No. Doesn't type check. M is a DFA, not a regular expression.

(e) Is $\langle M\rangle \in E_{DFA}$?

No. $L(M) \neq \emptyset$, $0 \in L(M)$.

(f) Is $\langle M, M\rangle \in EQ_{DFA}$?

Yes. $L(M) = L(M)$.

4.3 $ALL_{DFA} = \{ \langle A \rangle \mid A \text{ is a DFA } \& \ L(A) = \Sigma^* \}$.

Show that ALL_{DFA} is decidable.

$L(A) = \Sigma^* \rightarrow \overline{L(A)} = \phi$

Define Turing Machine M =

"On input $\langle A \rangle$, where A is a DFA,

1. Construct DFA $\overline{A}$
2. Run the T.M F which decides E_{DFA}, on input $\langle \overline{A} \rangle$.
3. If F accepts, accept. If F rejects, reject."

M is a decider because F is a decider, & M accepts $\langle A \rangle$ iff $\overline{L(A)} = \phi$ i.e $L(A) = \Sigma^*$. M decides ALL_{DFA}, so ALL_{DFA} is decidable.

(4.4) $A_{\epsilon CFG} = \{ \langle A \rangle \mid A \text{ is a CFG that generates } \epsilon \}$

Show that $A_{\epsilon CFG}$ is decidable.

Define Turing Machine M =

"On input $\langle A \rangle$, where A is a CFG

1. Run the T.M F which decides A_{CFG}, on input $\langle A, \epsilon \rangle$.
2. If F accepts, accept. If F rejects, reject."

M is a decider because F is a decider, & M accepts $\langle A \rangle$ iff A_{CFG} accepts $\langle A, \epsilon \rangle$ (i.e A generates ϵ). M decides $A_{\epsilon CFG}$.

$\therefore$ $A_{\epsilon CFG}$ is decidable.

(4.5) $E_{TM} = \{ \langle M \rangle \mid M \text{ is a T.M}, \; L(M) = \phi \}$.

Show that $\overline{E_{TM}}$ is recognizable.

$\overline{E_{TM}} = \{ \langle M \rangle \mid M \text{ is a T.M}, \; L(M) \neq \phi \}$

Define T.M N = " On input $\langle M \rangle$, where M is a T.M

1. For $i = 1, 2, 3, \ldots$

2. Run M for up to i steps on strings $s_1, s_2, \ldots s_i$

(the first i strings over the alphabet in shortlex order).

3. If M accepts, accept."

On input $\langle M \rangle$, if $L(M) \neq \phi$, there exists i, j such that s_i is accepted by M in j steps. N will eventually run M on input s_i for j steps & accept.

If $L(M) = \phi$, then M does not accept any strings, so N simply loops, since no value of i will result in M accepting some string.

$\therefore$ N recognizes $\overline{E_{TM}}$, so $\overline{E_{TM}}$ is recognizable.

(4.30) Let $A = \{ \langle M_1 \rangle, \langle M_2 \rangle, \ldots \}$ be a Turing-recognizable language, where every M_i is a decider. Show that there exists a decidable language D not decided by any of the deciders M_i.

We prove this by constructing a decidable language using diagonalization.

let $S = \{s_0, s_1, \ldots\}$ be the shortlex order of strings over the alphabet Σ.

Observation: Since A is recognizable, there is some enumerator E that enumerates A.

Construct T.M T = "On input w

1. let i be the index of w in S (i.e $w = s_i$)
2. Use E to obtain $\langle M_i \rangle$.
3. Run M_i on input w.
4. If M_i accepts, reject. If M_i rejects, accept."

T is a decider because each M_i is a decider.

However $\langle T \rangle$ doesn't appear in A because T differs from every M_i on at least one input - s_i. $\therefore$ $L(T)$ is a decidable language not decided by any M_i.