自動機與形式語言 Homework 4

Class 02 B03902086 李鈺昇

(1)

L1

Set $q_0 = q_{acc}$ so that the Turing machine directly accepts any input.

L2

Set $q_0 = q_{rej}$ so that the Turing machine directly rejects any input.

L3

First scan through the input to ensure (1) it's non-empty, (2) it only contains 0 or 1, otherwise reject. Then check the rightmost symbol of the input: If it is a 0, accept; otherwise, reject.

L4

Use a 3-tape (containing tape 1, 2, 3) TM to decide L4. Assume that on input w, tape 1 contains w, and other tapes are empty.

1. On input w, scan through tape 1 to ensure $w \in \{0, 1\}^* - \{\epsilon\}$, otherwise reject.
2. Assume $w = \lfloor n \rfloor$, the binary representation of n. If $n = 1$, reject.
3. Put $\lfloor 2 \rfloor$ on tape 2.
4. Let i be the number on tape 2 now. If $i = n$, accept; otherwise, do the following to check if $n \mod i = 0$:
   - Copy $\lfloor n \rfloor$ to tape 3.
   - While the number on tape 3 is non-negative, subtract i from tape 3.
   - After the loop, it's clear that $n \mod i = 0$ iff the number on tape 3 is 0.
5. If $n \mod i = 0$, reject; otherwise, add 1 to the number on tape 2 and go back to step 4.

(2)

Let $L_1 = \{0^n \mid n \geq 0\}$, $L_2 = \{1^n \mid n \geq 0\}$. Since L_1 and L_2 can be generated by regex (0^* and 1^*, respectively), they are regular and thus decidable. Let M_1 be the decider of L_1, M_2 be the decider of L_2.
Now if there is a parallel universe, M_1 decides L; otherwise M_2 decides L. Only one condition would be true, and in either case we know there **exists** a TM that decides L. So L is decidable.

(3)

We can first make the following table by definition:

	$\exists$ acc run, $\nexists$ rej run	$\nexists$ acc run, $\exists$ rej run	$\exists$ acc run, $\exists$ rej run	$\nexists$ acc run, $\nexists$ rej run
Original NTM	accept	reject	accept	reject
Bob's NTM	accept	reject	?	?

For a TM $\mathcal{M}$ and an input w,

- On one hand, if there are both accepting run(s) and rejecting run(s), then in the original version $\mathcal{M}$ accepts w, while in Bob's version $\mathcal{M}$ does not accept w. The behavior is not even undefined.

- On the other hand, if there is neither accepting run nor rejecting run, then in the original version $\mathcal{M}$ rejects w, while in Bob's version $\mathcal{M}$ does not reject w. The behavior is not even undefined.

Hence Bob's definition may lead to some confusion in the above situations.

(4)

Restate this problem as $\textsf{CFL-DFA-Equality} = \{\lfloor \mathcal{G}, \mathcal{A} \rfloor \mid \mathcal{G}$ is CFG, $\mathcal{A}$ is DFA, and $L(\mathcal{G}) = L(\mathcal{A})\}$.
First we know that $\textsf{CFL-Universality} = \{\lfloor \mathcal{G} \rfloor \mid \mathcal{G} = \langle \Sigma, V, R, S \rangle$ is CFG and $L(\mathcal{G}) = \Sigma^*\}$ is undecidable.
Let $\mathcal{A}_0$ be the trivial DFA that accepts all strings, i.e. $L(\mathcal{A}_0) = \Sigma^*$.

Now define the computable function $F : \Sigma^* \to \Sigma^*$ such that $F : \lfloor \mathcal{G} \rfloor \mapsto \lfloor \mathcal{G}, \mathcal{A}_0 \rfloor$. Since $\lfloor \mathcal{G} \rfloor \in \textsf{CFL-Universality}$ if and only if $L(\mathcal{G}) = \Sigma^* = L(\mathcal{A}_0)$ if and only if $\lfloor \mathcal{G}, \mathcal{A}_0 \rfloor \in \textsf{CFL-DFA-Equality}$, we know $\textsf{CFL-Universality} \leq_m \textsf{CFL-DFA-Equality}$. It follows that CFL-DFA-Equality is undecidable since CFL-Universality is undecidable.

(5)

Restate this problem as $\textsf{CFL-Complement} = \{\lfloor \mathcal{G} \rfloor \mid \mathcal{G} = \langle \Sigma, V, R, S \rangle$ is CFG and $\Sigma^* - L(\mathcal{G})$is CFL$\}$.
From (4) we know that $\textsf{CFL-DFA-Equality} = \{\lfloor \mathcal{G}, \mathcal{A} \rfloor \mid \mathcal{G}$ is CFG, $\mathcal{A}$ is DFA, and $L(\mathcal{G}) = L(\mathcal{A})\}$ is undecidable. Since DFA can be trivially transformed to deterministic pushdown automaton (DPDA), it follows that $\textsf{CFL-DPDA-Equality} = \{\lfloor \mathcal{G}, \mathcal{D} \rfloor \mid \mathcal{G}$ is CFG, $\mathcal{D}$ is DPDA, and $L(\mathcal{G}) = L(\mathcal{D})\}$ is undecidable.

Suppose CFL-Complement is decidable by TM $\mathcal{M}$. Then we can use the following procedure to decide CFL-DPDA-Equality:

- Given CFG $\mathcal{G}$ and DPDA $\mathcal{D}$, run $\mathcal{M}$ on $\lfloor \mathcal{G} \rfloor$.

- If the output of $\mathcal{M}$ is false, then $L(\mathcal{G})$ is not DCFL (as shown in the textbook). Then $L(\mathcal{G})$ cannot be equal to $L(\mathcal{D})$, so output false. Otherwise, if the output of $\mathcal{M}$ is true, then convert $\mathcal{G}$ to a DCFG $\mathcal{G}_{\mathcal{D}}$. Since the equality problem of DCFL, $\textsf{DCFL-Equality} = \{\lfloor L_1, L_2 \rfloor \mid L_1, L_2$ are DCFL, and $L_1 = L_2\}$, is decidable, say by TM $\mathcal{N}$, we can output the result of $\mathcal{N}$ on $\lfloor L(\mathcal{D}), L(\mathcal{G}_{\mathcal{D}}) \rfloor$.

Since CFL-DPDA-Equality is undecidable, we know CFL-Complement is undecidable.