COMP2210: Theory of Computing

Lecture 17

Decidable and Undecidable Problems

Corina Cirstea
Objectives for Today

- Undecidability of the Membership Problem for TMs
- Examples of Decidable and Undecidable Problems
Membership problem

- We have shown, using diagonalisation, that HP is undecidable.

- Q. What about

\[ MP \overset{\text{def}}{=} \{ M \# x \mid x \in L(M) \} \]?

- A. MP is undecidable.

- Proof: We could use diagonalisation (see Tutorial 7), but there is another way!
The proof, 1

• If MP is decidable, there is a total TM $K$ for it;

• We can use $K$ to solve HP. We will construct a new TM $N$.

• $N$ on input $M\#x$:

  1. constructs $M'$ that is like $M$ but with a new accept state and transitions from old accept and reject states to the new accept state. So whenever $M$ accepts or rejects, $M'$ accepts.

  2. simulates $K$ on $M'\#x$; if $K$ accepts then accept, if $K$ rejects then reject.

\[ M\#x \in L(N) \iff M'\#x \in L(K) \iff M \text{ halts on } x \]

• so $N$ decides HP!
The proof, 2

• we used a TM $K$ for $MP$ in order to solve $HP$;
• this proves that:

\[
MP \text{ decidable} \implies HP \text{ decidable}
\]

• but we know that $HP$ is undecidable, so $MP$ is undecidable;
• we have \textit{reduced} $HP$ to $MP$ (“$HP \leq MP$”);
• \textbf{Intuition}: $MP$ is at least as hard as $HP$; \textit{i.e.} by solving $MP$ we can solve $HP$.  

COMP 2210: Lecture 17 – p. 5
Problems Concerning Turing Machines

Is it decidable whether a given TM

- has at least 400 states?
- takes more than 400 steps on input $\epsilon$?
- takes more than 400 steps on some input?
- takes more than 400 steps on all inputs?
- ever moves its head more than 400 tape cells away from the left end marker on input $\epsilon$?
Problems Concerning Turing Machines (Cont’d)

Is it decidable whether a given TM

- accepts the null string $\epsilon$?
- accepts any string at all?
- accepts every string?
- accepts a given finite language?
- accepts a given regular language?
- accepts a given context-free language?
- accepts a given recursive language?
Problems Concerning Turing Machines

Is it decidable whether a given TM

• has at least 400 states?
  ○ A. Yes. We can read an encoding of the machine, and count (e.g. using special states) whether there are at least 400 states.

• takes more than 400 steps on input $\epsilon$?
  ○ A. Yes. We can simulate the running of the machine on $\epsilon$ and count the number of steps it takes, stopping after 400 steps.

• takes more than 400 steps on some input?
  ○ A. Yes. We cannot simulate for all inputs, but simulating for inputs of length up to 400 is sufficient, as in 400 steps no machine can explore more than 400 tape cells.

• takes more than 400 steps on all inputs?
  ○ A. Yes. Same idea as above.
Problems Concerning Turing Machines

- ever moves its head more than 400 tape cells away from the left end marker on input \( \epsilon \)?

A. Yes. If the machine never moves its head more than 400 tape cells away from the left end marker, then it will always move between configurations of the form \((q, y_1 \ldots y_{400}, h)\) with \(0 \leq h \leq 400\). Let \(N\) be the number of all possible configurations of this form. Simulating for \(N + 1\) steps may give two results:
  - the machine moves beyond the first 400 tape cells. In this case accept.
  - the machine never moves beyond the first 400 tape cells. In this case it must repeat a configuration (as only \(N\) configurations are possible), and from that point it will keep repeating transitions it already made. In this case reject.

(detailed answers in Kozen, pp 235-236)
Problems Concerning Turing Machines

- accepts the null string $\epsilon$?
  A. No (see next lecture).

- accepts any string at all?
  A. No.

- accepts every string?
  A. No.

- accepts a given finite language?
  A. No.

- accepts a given regular language?
  A. No (see Tutorial 7).

- accepts a given context-free language?
  A. No.

- accepts a given recursive language?
  A. No.

(detailed answers in Kozen, pp 237-238)