More undecidable problems

Deepak D'Souza

Department of Computer Science and Automation
Indian Institute of Science, Bangalore.

20 November 2018

Outline

More decidable/undecidable problems

Problem (a)

Is it decidable whether a given Turing machine has at least 481 states? Assume that the TM is given using the encoding below:

$$0^n 1 0^m 1 0^k 1 0^s 1 0^t 1 0^r 1 0^u 1 0^v \; 1 \; 0^p 1 0^a 1 0^q 1 0^b 1 0 \; 1 \; 0^{p'} 1 0^{a'} 1 0^{q'} 1 0^{b'} 1 00 \; \cdots 1 \; 0^{p''} 1 0^{a''} 1 0^{q''} 1 0^{b''} 1 0.$$

000100001001010010001000010000 1 01000101000100 1 0100100100100 1 010101010.

More decidable/undecidable problems

Problem (a)

Is it decidable whether a given Turing machine has at least 481 states? Assume that the TM is given using the encoding below:

$$0^n10^m10^k10^s10^t10^r10^u10^v\ 1\ 0^p10^a10^q10^b10\ 1\ 0^{p'}10^{a'}10^{q'}10^{b'}100\ \cdots 1\ 0^{p''}10^{a''}10^{q''}10^{b''}10.$$

$$0001000010010100100010001000 0\ 1\ 01000101000100\ 1\ 0100100100100\ 1\ 010101010.$$

Yes, it is.

We can give a TM N which given $enc(M)$

- Counts the number of states in M upto 481.
- Accepts if it reaches 481, rejects otherwise.

More decidable/undecidable problems

Problem (b)

Is it decidable whether a given Turing machine takes more than 481 steps on input ϵ without halting?

0001000010010100100010001000 1 01000101000100 1 0100100100100 1 010101010.

More decidable/undecidable problems

Problem (b)

Is it decidable whether a given Turing machine takes more than 481 steps on input ϵ without halting?

0001000010010101001000100010000 1 01000101000100 1 0100100100100 1 010101010.

Yes, it is.

We can give a TM N which given $enc(M)$

- Uses 4 tapes: On the 4th tape it writes 481 0's.
- Uses the first 3 tapes to simulate M on input ϵ, like the universal TM U.
- Blanks out a 0 from 4th tape for each 1-step simulation done by U.
- Rejects if M halts before all 0's are blanked out on 4th tape, accepts otherwise.

More decidable/undecidable problems

Problem (c)

Is it decidable whether a given Turing machine takes more than 481 steps on *some* input without halting?

0001000010010100100010001000 1 01000101000100 1 0100100100100 1 010101010.

More decidable/undecidable problems

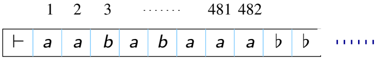
Problem (c)

Is it decidable whether a given Turing machine takes more than 481 steps on *some* input without halting?

0001000010010100100010001000010000 1 01000101000100 1 0100100100100 1 010101010.

Yes, it is.

Check if M runs for more than 481 steps on some input x of length upto 481. If so accept, else reject.



More decidable/undecidable problems

Problem (d)

Is it decidable whether a given Turing machine takes more than 481 steps on *all* inputs without halting?

0001000010010100100010001000 0 1 01000101000100 1 0100100100100 1 010101010.

More decidable/undecidable problems

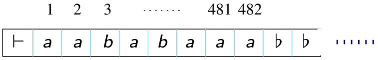
Problem (d)

Is it decidable whether a given Turing machine takes more than 481 steps on *all* inputs without halting?

0001000010010100100010001000010000 1 01000101000100 1 0100100100100 1 010101010.

Yes, it is.

Check if M runs for more than 481 steps on each input x of length upto 481. If so accept, else reject.



More decidable/undecidable problems

Problem (e)

Is it decidable whether a given Turing machine moves its head more than 481 cells away from the left-end marker, on input ϵ?

0001000010010100100010001000 1 01000101000100 1 0100100100100 1 010101010.

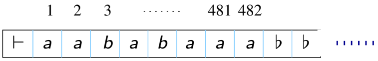
More decidable/undecidable problems

Problem (e)

Is it decidable whether a given Turing machine moves its head more than 481 cells away from the left-end marker, on input ϵ?

00010000100101001000100010000 1 01000101000100 1 0100100100100 1 010101010.

Yes, it is.
Simulate M on ϵ for upto $m^{481} \cdot 482 \cdot k + 1$ steps. If M visits the 482nd cell, accept, else reject.



More decidable/undecidable problems

Problem (f)

Is it decidable whether a given Turing machine accepts the null-string ϵ?

More decidable/undecidable problems

Problem (f)

Is it decidable whether a given Turing machine accepts the null-string ϵ?

No.
If it were decidable, say by a TM N, then we could use N to decide HP as follows: Define a new machine N' which given input $M\#x$, outputs the description of a machine $M'_{M,x}$ which:

- erases its input
- writes x on its input tape
- Behaves like M on x
- Accepts if M halts on x.

N' then calls N with input $M'_{M,x}$.
N accepts $M'_{M,x}$ iff $M'_{M,x}$ accepts ϵ iff M halts on x.

Turing machine M' for Problem (f)

$$L(M'_{M,x}) = \begin{cases} A^* & \text{if } M \text{ halts on } x \\ \emptyset & \text{if } M \text{ does not halt on } x. \end{cases}$$

More decidable/undecidable problems

Problem (g)

Is it decidable whether a given Turing machine accepts any string at all? That is, is $L(M) \neq \emptyset$?

More decidable/undecidable problems

Problem (h)

Is it decidable whether a given Turing machine accepts all strings? That is, is $L(M) = A^*$?

More decidable/undecidable problems

Problem (i)

Is it decidable whether a given Turing machine accepts a finite set?

More decidable/undecidable problems

Problem (j)

Is it decidable whether a given Turing machine accepts a regular set?

More decidable/undecidable problems

Problem (j)

Is it decidable whether a given Turing machine accepts a regular set?

Given M and x, build a new machine M' that behaves as follows:

1. Saves its input y on tape 2.
2. writes x on tape 1.
3. runs as M on x.
4. if M gets into a halting state, then
   - M' takes back control,
   - Runs as M_R on y,
   - (Here M_R is any TM that accepts a non-regular language R, say $R = \{a^n b^n \mid n \geq 0\}$).
   - M' accepts iff M_R accepts.

Turing machine M' for Problem (j)

$$L(M') = \begin{cases} R & \text{if } M \text{ halts on } x \\ \emptyset & \text{if } M \text{ does not halt on } x. \end{cases}$$

More decidable/undecidable problems

Problem (k)

Is it decidable whether a given Turing machine accepts a CFL?

More decidable/undecidable problems

Problem (I)

Is it decidable whether a given Turing machine accepts a recursive set?