CPSC 421/501 Nov 23

- Midterms should be back Wed-Thu (Fri?)
- Today: most difficult topic: **ABSTRACTIFY LAST CLASS**
- Next 4 lectures after today:
  easy stuff; you can read textbook
  if you need to finish other projects:

P vs NP [Sip] Ch 7

How to solve / not to solve
P vs NP [Sip] Ch 9

(current ideas, as of 1970's)

Start next Thursday, Nov 25

Tues, Nov 30

MAIN POINT: COOK-LEVIN THEOREM

AND ITS PROOF, ORACLE MACHINES

=

Question:

- What if quantum computer
- " " oracle TM
- " " add power in 2323?
- " " we allow non-halting computations with some convention [Spencer Bosma]
- etc.

=

What is ~~the~~ a reasonably small amount of ~~minimal~~ stuff we need to assume to give a proof that ACCEPTANCE, HALT undecidable, etc.

One way to set things up:

Set "programs" $\mathcal{P}$

" "inputs" $\mathcal{I}$

Result: $\mathcal{P} \times \mathcal{I} \to \{\text{yes}, \text{no}, \text{loops}\}$

i.e. $\{\text{accept}, \text{reject}, \text{loops}\}$

Say (think of $\mathcal{I} = \text{ASCII}^*$)

EncodeProg: $\mathcal{P} \to \mathcal{I}$ (injective, although not nec.)

EncodeBoth: $\mathcal{P} \times \mathcal{I} \to \mathcal{I}$ (injective not nec)

$\mathcal{S} = (\mathcal{P}_{\mathcal{S}}, \mathcal{I}_{\mathcal{S}}, \text{Result}_{\mathcal{S}}, \text{EncodeProg}_{\mathcal{S}}, \text{EncodeBoth}_{\mathcal{S}})$

Rem: In {Sip}

A_{DFA}, A_{NFA}, A_{TM}

i.e.,

ACCEPTANCE$_{DFA}$, --- NFA, -- TM

Here $\mathcal{L}$ = DFA, (context of TM)

= NFA ~~blah~~

= TM blah

=

Need:

the "ability to negate"

i.e.,

define $\neg$ yes $\overset{def}{=}$ no

$\neg$ no $\overset{def}{=}$ yes

$\neg$ loops $\overset{def}{=}$ loops

Axiom / Property (Part of definition)

$\forall p \in \mathcal{P}, \quad \exists p' \in \mathcal{P}$ s.t. ("expressive P-I settings")

$$\boxed{\forall i \in \mathcal{I}, \quad \text{Result}(p', i) = \neg \text{Result}(p, i)}$$

Want to be able to preprocess the input in various ways. →

We need!

$\forall p \in \mathcal{P}, \quad \exists p' \in \mathcal{P}$ s.t.

$$\boxed{\begin{array}{l} \forall i \in \mathcal{I} \\ \text{Result}(p', \quad \text{blah}) = \text{Result}(p, i) \end{array}}$$

you want to "feed a program description

$\langle \varphi \rangle \rightsquigarrow \langle \varphi, \langle \varphi \rangle \rangle$

description of φ

description of φ with $\langle \varphi \rangle$ tacked on at the end

as preprocessing!

$\forall p \in \mathcal{P} \quad \exists p' \in \mathcal{P}$ s.t.

use d, but no explicit elt of d

$$\boxed{\begin{aligned} &\forall q \in \mathcal{P} \quad \text{Result}(p', \underline{\text{EncodeProg}}(q)) \\ &= \text{Result}(p, \underline{\text{EncodeBoth}}(q, \underline{\text{EncodeProg}}(q))) \end{aligned}}$$

i.e., $\langle\ \rangle$ = description!

$\forall q$

$$\underbrace{p'(\langle q\rangle)}_{\text{run } p' \text{ on description of } q} = p(\langle q, \langle p\rangle\rangle)$$

For us:

Standardized TM

$$101 \# 1111 \# 111001 \# \underbrace{\sim\sim}_{\delta \text{ function}}$$

Break 5 min 10:18 - 10:23

Sipser in Ch 4 talks about

undecidable ...

unrecognizable ...

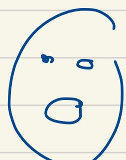
only in the context of

classical TM's ...

but in
Ch 6 ...
Ch 9 ...

Imagine: you have a calculator

with a $\boxed{\text{sin}}$ $\boxed{\text{cos}}$

$\boxed{\text{HALT}}$ ← 

Maybe in 2323, Scifi ;)

idea from logic, philo...

So now say!

classical TM + oracle for _ _

whatever

HALT

Sci fi
232323

thing you can do in Q.C _ _

=

Claim: If you have a whatever button/oracle on you

TM
C
Javascript

Futuristic Prog Lgs

then

$$\text{HALT}_{\text{oracle } \boxed{\text{whatever}} + \text{TM}}$$

is undecidable if you have

$$\text{oracle } \boxed{\text{whatever}} + \text{TM}$$

Proof!

① Write down all of [S, p], §4.2

proof that ACCEPTANCE in

context of - - . Then go to Ch 5.

- - - ——

OR

② Check the axioms for expressive P-I system

+ universal machine - -

Theorem: For any oracle f (see [Sip] my notes uncomputability)

$HALT_{\text{oracle } f + TM}$

is undecidable if you have

oracle f + TM

Pf: Verify some "~~simple~~ minimal" axioms (with minimal work)

$\Longrightarrow$

So HALT is undecidable for TM's

" $HALT_{\text{oracle } HALT}$ " " " TM's with oracle HALT

[Sip] Ch 4.1

$\text{ACCEPTANCE}_{\text{DFA}}$ can be solved by TM's

$\text{ACCEPTANCE}_{\text{NFA}}$ " " " " "

but

$\text{ACCEPTANCE}_{\text{TM}}$

HALT_{TM} can't be ~~solved~~ decided by TM

(they can be recognized by a univ TM)

Usual proof is "by contradiction..."

For us:

If you had a Univ TM that ~~were~~ was a decider, then you could build a delightful TM

i.e. a $d \in P$ s.t.

$\forall q \in P$

$\mathrm{Result}(d, q) =$

$\neg \mathrm{Result}(u, \ldots)$

$\neg \mathrm{Result}(q, \langle q \rangle)$