# More reductions

# &

# Rice's Theorem

# Recap end of LBA discussion

$E_{LBA}$ undecidable

Can build an LBA

$B_{M,w}$ That check

if input is a CH

for $M$ on input $w$

If $R$ decide $E_{LBA}$ Then

$R(\langle B_{M,w} \rangle)$ accepts $\Longleftrightarrow$

$L(B_{M,w}) = \emptyset \Longleftrightarrow$

$M$ doesn't accept $w$

$ALL_{CFG}$ is undecidable

→ Can Build a CF grammar $G_{M,w}$

s.t. $L(G)$ is all strings

not CH for M on w

Let R recognizes $ALL_{CFG}$

then $R(\langle G_{M,w} \rangle)$ accepts

$\Longleftrightarrow L(G_{M,w}) = \Sigma^*$

$\Longleftrightarrow$ Every string is

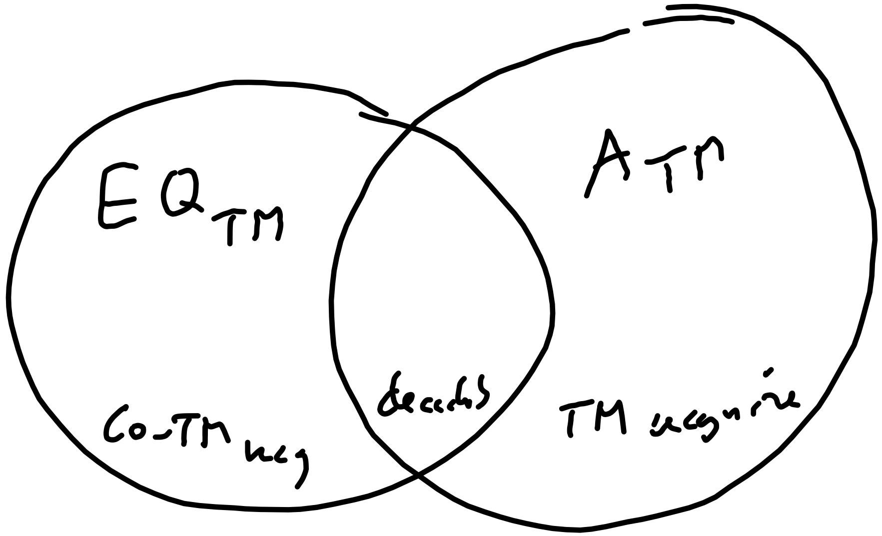
not a CH for M on w

$\Longleftrightarrow \nexists$ CH for M on w

$\Longleftrightarrow$ M doesn't accept w

(*) How to reverse alternate strings in C++

# Terminology




Co-Turing recognizable

= complement of L is Turing recognizable

$EQ_{TM}$ is not recognizable

but $\overline{EQ_{TM}}$ is

Point #1

input doesn't matter much

$HALT_{TM} = \{\langle M, w\rangle : M \text{ halts on } w\}$

$HALT_{UIUC} = \{\langle M\rangle : M \text{ halts on input "UIUC"}\}$

Prove that not decidable.

Assume R decides $HALT_{UIUC}$

# Decider for $A_{TM}$

- input = $\langle M, w \rangle$
- build new TM $M_w$
  - input = $x$ (ignore)
  - Simulate $M$ on $w$
  - if $M$ accepts $w$, accept
  - if $M$ rejects $w$, reject
- $R(\langle M_w \rangle)$
- if yes, actually run $M_w$
  ~~to~~ decide if $M$ accepts $w$
- if no, reject

Point #2

Lots of reductions have same outline

Top-level outline

Suppose $R$ decides $L$

Make decider for $A_{TM}$

- input $\langle M, w \rangle$
- build new TM $M'_w$
- run $R(\langle M'_w \rangle)$
- return simple permutation of result of $R(\langle M'_w \rangle)$

$M'_w$ looks like

(a) • input $x$

• If $x$ has easily tested property $P$

accept $x$

• Else simulate $M$ on $w$

e.g. $x \in 0^n 1^n$

on input $x$

$L(M'_w) = \{$ – what $P$ tests $0^n 1^n$

$\{$ – $\Sigma^*$

(b) $M'_w$ looks like

- input $x$
- Run $M$ on $w$
- if $M$ rejects, reject
- Else accept $x$ iff
  $x$ satisfies $P$

easily tested

$L(M'_w) = \emptyset$ if $M$ doesn't accept $w$

or what $P$ tests if $M$ accepts $w$

Example

$$L_3 = \{ \langle M \rangle : |L(M)| = 3 \}$$

$M'_w$ = • input $x$

• simulate $M$ on $w$

• If $M$ rejects, we reject

• Else accept $x$ iff

$x \in \{$ "iowa", "purdue", "michigan" $\}$

$L(M'_w)$ = - if $M$ doesn't accept $w$

$= \phi$

- if $M$ accepts $w$

$= \{$ "iowa", "purdue", "michigan" $\}$

Title: Apr 19-11:34 AM (11 of 18)

# Rice's Theorem

if P is a set of TM's s.t.

① P is "non-trivial"

i.e. $P \neq \emptyset$

and $P \neq$ all TM's

② Membership in P depends only on TM's language

if $L(M_1) = L(M_2)$

then $M_1 \in P \iff M_2 \in P$

then P is undecidable.

Proof: Suppose $N$ decides $P$

Need to construct a TM

decides $A_{TM}$

Suppose WLOG *That

$T_\emptyset \notin P$

$T_\emptyset$ = a TM s.t. $L(T_\emptyset) = \emptyset$

Pick $T \in P$ Because $P$ non-trivial

(*) $\mathcal{P}$ is decidable
iff $\overline{\mathcal{P}}$ is decidable

So if $T_\phi \in \mathcal{P}$, do proof
using $\overline{\mathcal{P}}$

# Decider for $A_{TM}$

- input $\langle M, w \rangle$
- Build new TM $M'_w$
  - input $x$
  - Simulate $M$ on $w$
  - if $M$ rejects, reject
  - Else run $T$ on $x$ and do what $T$ does
- Run $N(\langle M'_w \rangle)$

behaves like $T_\emptyset$ if $M$ doesn't accept $w$

---

$L(M'_w) = \emptyset = L(T_\emptyset)$ if $M$ doesn't accept $w$

$L(T)$ if $M$ does accept $w$

By (2) This means That $N(\langle M'_w \rangle)$ accepts iff $M'_w \in P$

iff $L(M'_w) = L(T)$

iff $M$ accepts $w$

# Types of TM decision problems

- TM language properties

  $\Longrightarrow$ undecidable

- TM state diagram (TM has 13 states)

  $\Longrightarrow$ decidable

- TM behavior properties

  $\Longrightarrow$ ???

$L_x = \{ \langle M \rangle :$ M writes an x onto the tape on some move on empty input $\}$

undecidable

$L_{left} = \{ \langle M \rangle :$ M never moves left on empty input $\}$

decidable

$L_x$ is not decidable

Let $R$ decide $L_x$

Decider for $A_{TM}$

- input = $\langle M, w \rangle$
- Build new TM $M'_w$
  - input = $x$ (ignored)
  - Simulate $M$ on $w$
    but use $x'$ in place of $x$
    everywhere
  - if $M$ accepts $w$, write $x$
    onto tape
  - if $M$ rejects $w$, reject

- $R(\langle M'_w \rangle)$