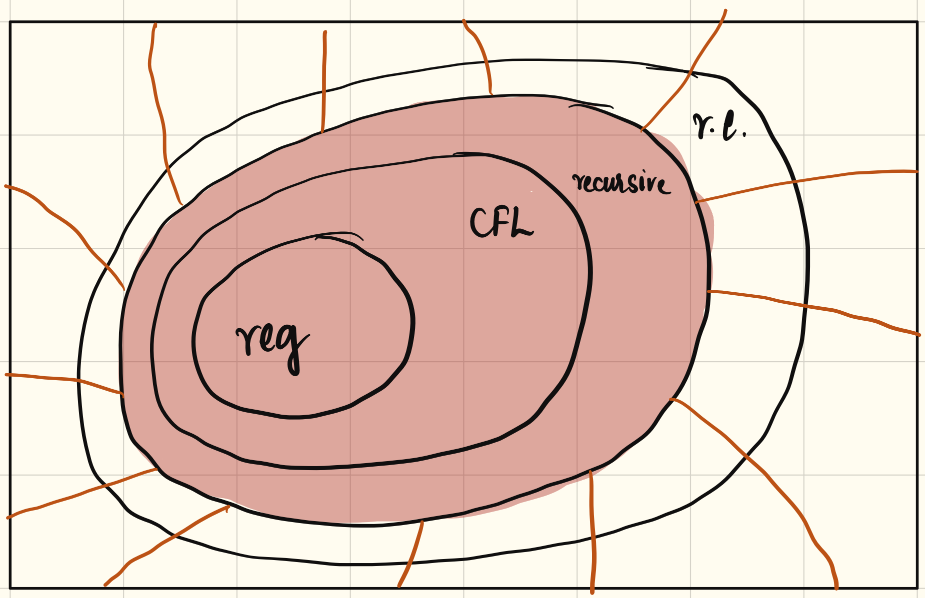
07 Apr 2025 – Theory of Computation – Week 13

$MP := \{ \langle M, x \rangle \mid M \text{ accepts } x \}$

<u>Thm:</u> MP is undecidable

$\iff$ MP is not recursive

$\iff$ No <u>halting</u> TM that recognizes MP. (proof using diagonalization)



Thm: If A is r.e. and $\bar{A}$ is r.e. then A is recursive.

(run simultaneously)

M_A accepts A

$M_{\bar{A}}$ accepts $\bar{A}$

Q: $\overline{MP}$ → recursive? ✗ recursive languages are closed under ($\bar{\ }$)

↳ swap accept & reject states

(proof by contradiction: $\overline{\overline{MP}} = MP$ is recursive ⇒⇐)

→ r.e? ✗ Suppose

$\overline{MP} \in re$

$MP \in re$ ⇝ construct TM simulate ⟨M, x⟩

$\underbrace{\qquad\qquad}$

↓

MP is recursive ⇒⇐

→ neither ✓

Halting Problem

$$HP = \{ \langle M, x \rangle \mid M \text{ halts on } x \}$$

Thm: HP is undecidable.

Proof: Suppose HP is decidable $\Rightarrow$ there is a halting TM that recognizes HP.

Let M_{HP} be a halting TM recognizing HP.

Goal: Use M_{HP} to decide MP (hence, contradiction)

Let $\langle M, x \rangle$ be an input to MP.

Let T be a turing machine that we design as follows:

On input $\langle M, x \rangle$

i) Run M_{HP} on $\langle M, x \rangle$

If ans of M_{HP} = does not halt

then reject

else

simulate M on x

and answer accordingly.

M_{HP} will always halt by assumption

Clearly T is a halting TM.
and language recognized by T is MP

$\Rightarrow\Leftarrow$

Two techniques

1) diagonalization (first principles)

2) use other problems:

We assume a problem B is "solvable". $\rightarrow$ decidable.
Use it to solve another problem A that cannot be solved.

A is "reducible" to B

MP is "reducible" to HP } shown by our proof

Lemma: If A is reducible to B and B is decidable. Then A is decidable.

Contrapositive: If A is undecidable, then B is undecidable

A reducible to B.

Emptiness Problem

$$B := \{ \langle M \rangle \mid L(M) = \phi \}$$

Show B is undecidable.

Proof: Assume B is decidable. $\Longrightarrow$ $\exists$ a halting TM M_B that decides B.

Solve MP using B.

M'' → for MP

$M'' :=$ On input $\langle M, x \rangle$

M'' constructs M'
(a description)

Construct M' as follows:

$M' :=$ on input y

1) Simulate M on x

2) if M accepts, then reject y.
else accept.

has to be fixed

$L(M') = \phi$ iff M accepts x

??

09 Apr 2025

- Undecidability
- MP, HP
- diagonalization
- reduction

  A reducible to B
  if we can solve A
  using solution to B

Thm: MP is reducible to HP

→ **Proof of undecidability using reduction**

Assume $\exists$ a halting TM for HP. Using this assumption you build a TM for MP.

Emptiness problem:

$$empty := \{ \langle M \rangle \mid M \text{ is TM and } L(M) = \phi \}$$

Thm: Empty is undecidable

Proof: We will reduce MP to empty.

Let M_1 be a turing machine that we design as follows:

On input $\langle M, x \rangle$

Assume that empty is decidable. Let E be a halting TM s.t. $L(E) = \text{empty}$.

What happens if $\langle M \rangle$ to E?

$$\text{Output of } E = \begin{cases} \text{Yes} & \text{if } L(M) = \phi \\ \text{No} & \text{if } L(M) \neq \phi \end{cases}$$

$\langle M, x \rangle$ will not be accepted

don't know

Construct M'

1) On input y → y is an input to M'

$y \in \Sigma^*$

2) If $y \neq x$ then reject
3) If $y = x$ then run M on x.
4) If M accepts, then M' accepts
5) If M rejects, then M' rejects.

if M accepts x, $L(M') = \{x\}$

if M does not accept x, $L(M') = \phi$

M' is a description ⇝ finite time to describe

we won't actually run M on x.

M_1 passes $\langle M' \rangle$ to E

If E accepts, M does not accept x, $\langle M, x \rangle \notin MP$.

If E rejects, $L(M') = \{x\} \Rightarrow \langle M, x \rangle \in MP$

M_1 is a halting TM.

M_1 solves MP

$\Rightarrow\Leftarrow$

M' depends on M and x

But it can always be built.

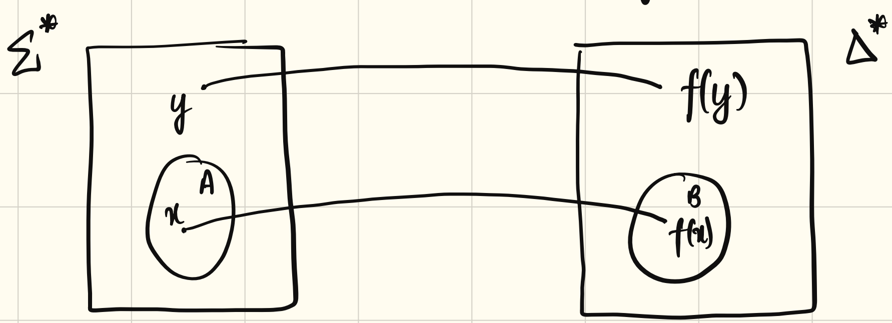
Def^n: Call a function $f : \Sigma^* \to \Delta^*$ computable if $\exists$ a halting TM s.t on input x, M writes $f(x)$ on its output tape and halts.

addition is a computable function.

A many-one reduces to B

Def^n: **Many-one reduction**. We say that $A \leq_m B$, $A \subseteq \Sigma^*$, $B \subseteq \Delta^*$ if $\exists$ a computable f^n f such that $\forall x \in \Sigma^*$,

$$x \in A \iff f(x) \in B$$



Note: f does not have to be one-one or many-one.

$$A \leq_m B \not\Rightarrow B \leq_m A$$

$$A \leq_m B \Rightarrow \overline{A} \leq_m \overline{B}$$

same f

$A \leq_m B$ and B is decidable

Given input x, decide $x \in A$:

Compute $f(x)$

Pass $f(x)$ to TM that decides B.

$f(x) \in B \Rightarrow x \in A$

$f(x) \notin B \Rightarrow x \notin B$

General reduction $\rightsquigarrow$ complement, etc.

Many-one reduction $\rightsquigarrow$ only one subroutine

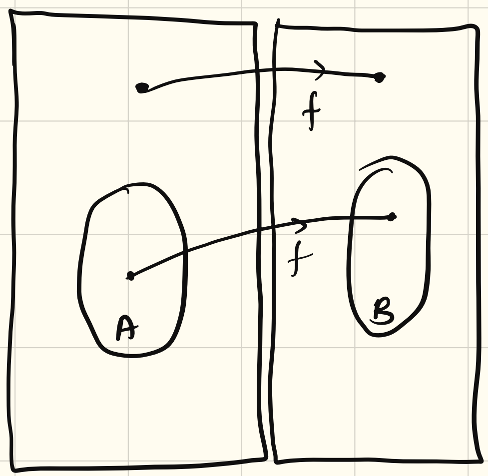
10 Apr 2025

→ Reductions

→ $A \leq_m B$ if $\exists$ a computable function f s.t

$\forall x \in \Sigma^*$

$$x \in A \iff f(x) \in B$$



" many-one / mapping reduction "

Obs: if $A \leq_m B$, then $\bar{A} \leq_m \bar{B}$.

Proof: $A \leq_m B$ via the function f.

The same f also shows that

$$\bar{A} \leq_m \bar{B}$$

$$x \notin A \iff f(x) \notin B$$

Lemma: If $A \leq_m B$ and B is decidable. Then, A is decidable.

Proof: Let $A \leq_m B$ via the reduction f.

Let M be a halting TM deciding B.

halt $\Rightarrow$ finite description

irrational no. cannot be printed

Construct M_1 which takes x as input.

On an input x

1) Compute $f(x)$
2) Run M on x.
3) Output accept or reject as M does.

M_1 is a halting TM that decides A.

$\blacksquare$

<u>Corollary</u>: If $A \leq_m B$ and A is undecidable, then B is undecidable.

$\rightsquigarrow$ If not, then A is decidable (by above lemma) $\Rightarrow\Leftarrow$

$\blacksquare$

Lemma 2: If $A \leq_m B$ and B is r.e. then A is r.e.

what if general reduction? "A reduces to B" make membership tests in B. Some modifications, then output.

Proof: Show that $\exists$ a TM M' that recognizes A.

Let f be a reduction showing $A \leq_m B$.
Let M_B be a TM recognizing B.

Construct M_A as follows:

On an input x:

1) Compute $f(x)$
2) Run M_B on $f(x)$ finite description
3) If M_B accepts $f(x)$, then M_A accepts x.

4) If M_B rejects $f(x)$, then M_B rejects x.

M_A recognizes A. ■

Corallary: If $A \leq_m B$ and A is not r.e. Then B is not r.e.

↳ If not, A is r.e., $\Rightarrow\Leftarrow$

Q: If "A reduces to B" via a general reduction and B is r.e. then what about A?

Procedure

< test for membership in B >

this goes on looping

< . . . >

this procedure may not depend on the answer of the first test

Obs: A reduces to B via a general reduction and B is an r.e. then A may or may not be r.e.

<u>Example</u>

M.P is r.e

$\overline{M.P}$ is not r.e

If r.e., then MP becomes recursive. $\Rightarrow\Leftarrow$

$\overline{MP}$ reduces to MP via the general reduction.

Why?

$$\langle M, x\rangle \in \overline{MP} \iff \langle M, x\rangle \in MP$$

Use TM for MP $\rightarrow$ negate the output

Q: Let $A \subseteq \Sigma^*$. Then does $\bar{A}$ reduce to A?

??

Contradictory. You have a TM for MP, you

Example:

$$\text{Empty} = \{ \langle M\rangle \mid M \text{ is a TM and } L(M) = \phi \}$$

Recall: Empty is undecidable.

Proof: Assume Empty is decidable with E be the halting TM that recognizes E.

Let M_1 be a TM that works as follows on an input $\langle M, x \rangle$.

1) Construct a machine M' which does the following on input y.

1) If $y \neq x$, reject
2) If $y = x$, run M on x
3) M' accepts if M accepts
4) M' rejects if M rejects

$$L(M') = \begin{cases} \{x\} & \text{if } M \text{ accepts } x \\ \phi & \text{o/w} \end{cases}$$

2) Run E on $\langle M' \rangle$

3) Answer accept if E rejects

reject if E accepts.

$\boxed{MP \leq_m \text{Empty} ?}$ X

$$\langle M, x \rangle \xrightarrow{1} \langle M' \rangle$$

$$\langle M, x \rangle \in MP \Rightarrow L(\langle M' \rangle) \neq \phi$$
$$\Rightarrow \langle M' \rangle \notin \text{Empty}$$
$$\langle M, x \rangle \notin MP \Rightarrow L(\langle M' \rangle) = \phi$$
$$\Rightarrow \langle M' \rangle \in \text{Empty}$$

<u>Note again</u>:

We never run M'. We only give a description of M'

~~Corollary~~

$\therefore$

$$MP \leq_m \overline{Empty}$$

or

$$\overline{MP} \leq_m Empty$$

We have seen :

Empty is undecidable

$\overline{Empty}$ is undecidable

Q : Empty $\in$ r.e ? $\longrightarrow$ NO

$\overline{MP}$ is not r.e.

corollary : $\overline{MP} \leq_m Empty$

$\therefore$ Empty is not even r.e.

$\overline{Empty}$ is also not r.e.