COMP30026
Models of Computation

Lecture 20: Undecidable Languages and Reducibility

Mak Nazecic-Andrlon and William Umboh

Semester 2, 2024

Some material from Michael Sipser's slides

Where are we?

Last time:

- Simulation and reduction
- Decidability of various problems about automata and grammars:
  E_{DFA} , A_{CFG} , E_{CFG}
- A_{TM} is T-recognizable
- Countable Sets

Today: (Sipser §4.2)

- A_{TM} is undecidable
- The diagonalization method
- $\overline{A_{\text{TM}}}$ is T-unrecognizable
- The reducibility method
- Halting Problem

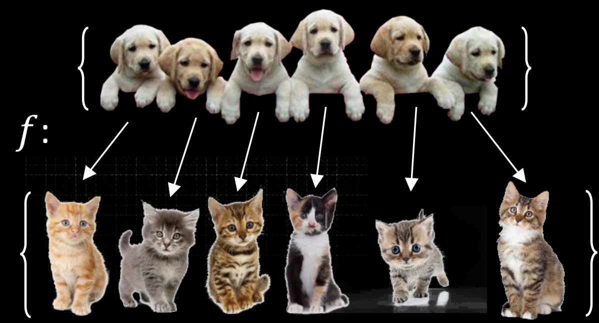
Recall: The Size of Infinity

How to compare the relative sizes of infinite sets?

Cantor (~1890s) had the following idea.

Informally, two sets have the same size if we can pair up their members.

Defn: Say that set A and B have the same size if there is
a $\underbrace{\text{one-to-one}}_{\substack{x \neq y \rightarrow \\ f(x) \neq f(y) \\ \text{“injective”}}}$ and $\underbrace{\text{onto}}_{\substack{\text{Range } (f) = B \\ \text{“surjective”}}}$ function $f: A \rightarrow B$



We call such an f a 1-1 correspondence

This definition works for finite sets.

Apply it to infinite sets too.

Recall: Countable Sets

Let $\mathbb{N} = \{1,2,3, \dots\}$ and let $\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}$

Show $\mathbb{N}$ and $\mathbb{Z}$ have the same size

$\mathbb{N}$ n	$f(n)$ $\mathbb{Z}$
1	0
2	-1
3	1
4	-2
5	2
6	-3
7	3
⋮	⋮

Let $\mathbb{Q}^+ = \{ {}^{m}/_{n} \mid m, n \in \mathbb{N}\}$

Show $\mathbb{N}$ and $\mathbb{Q}^+$ have the same size

$\mathbb{Q}^+$	1	2	3	4	…
1	1/1	1/2	1/3	1/4	
2	2/1	2/2	2/3	2/4	…
3	3/1	3/2	3/3	3/4	
4	4/1	4/2	4/3	4/4	
⋮		⋮			

$\mathbb{N}$ n	$f(n)$ $\mathbb{Q}^+$
1	1/1
2	2/1
3	1/2
4	3/1
5	3/2
6	2/3
7	1/3
⋮	⋮

Think of table as a grid graph and $f(n)$ is the n th number in BFS traversal starting from top-left corner

Defn: A set is countable if it is finite or it has the same size as $\mathbb{N}$.

Both $\mathbb{Z}$ and $\mathbb{Q}^+$ are countable.

$\mathbb{R}$ is Uncountable – Diagonalization

Let $\mathbb{R}$ = all real numbers (expressible by infinite decimal expansion)

Theorem: $\mathbb{R}$ is uncountable

Proof by contradiction via diagonalization: Assume $\mathbb{R}$ is countable

So there is a 1-1 correspondence $f: \mathbb{N} \to \mathbb{R}$

> **Check-in 20.1**
> Does the proof still work if we define x such that it differs from every real on the list in the first digit?

n	$f(n)$
1	
2	
3	
4	
5	
6	
7	
$\vdots$	

Diagonalization

Demonstrate a number $x \in \mathbb{R}$ that is missing from the list.

$$x = 0\,.$$

differs from the n^{th} number in the n^{th} digit
so cannot be the n^{th} number for any n.

Hence x is not paired with any n. It is missing from the list.

Therefore f is not a 1-1 correspondence.

$\mathbb{R}$ is Uncountable – Corollaries

Let $\mathcal{L}$ = all languages

Corollary 1: $\mathcal{L}$ is uncountable

Proof: There's a 1-1 correspondence from $\mathcal{L}$ to $\mathbb{R}$ so they are the same size.

Σ^*	$\{\varepsilon,$	0,	1,	00,	01,	10,	11,	000,	…
$A \in \mathcal{L}$	{	0,		00,	01,				…
$f(A)$	.0	1	0	1	1	0	0	0	…

Observation: $\Sigma^* = \{\varepsilon, 0,1,00,01,10,11,000, \ldots\}$ is countable.

Let $\mathcal{M}$ = all Turing machines

Observation: $\mathcal{M}$ is countable.

Because $\{\langle M\rangle \mid M$ is a TM$\} \subseteq \Sigma^*$.

Corollary 2: Some language is not recognizable.

Because there are more languages than TMs.

We will show some specific language A_{TM} is not decidable.

Consider the following mapping f from $\mathcal{M}$ to $\mathcal{L}$

$\mathcal{M}$ is countable so we can list all of $\mathcal{M}$

M	$f(M)$
⋮	⋮

$\mathcal{L}$ is uncountable so there is some language not in list!

A_{TM} is undecidable

Recall $A_{\mathrm{TM}} = \{\langle M, w\rangle |\ M \text{ is a TM and } M \text{ accepts } w\}$

Theorem: A_{TM} is not decidable

An unusual aspect of proof:

- Proof considers input of the form $\langle M, \langle M\rangle\rangle$
- $\langle M, \langle M\rangle\rangle$ is in A_{TM} if M accepts when it is given its own encoding

Example: Optimizing compiler

- Suppose we write an optimizing compiler in C
- First feed it to an unoptimized C compiler to get an unoptimized optimizing compiler
- Feeding the optimizing compiler's code to the unoptimized executable yields an optimized optimizing compiler!

A_{TM} is undecidable

Recall $A_{\mathrm{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$

Theorem: A_{TM} is not decidable

Proof by contradiction: Assume some TM H decides A_{TM}.

So H on $\langle M, w\rangle = \begin{cases} \textit{Accept} & \text{if } M \text{ accepts } w \\ \textit{Reject} & \text{if not} \end{cases}$

Use H to construct TM D

$D =$ "On input $\langle M\rangle$

1. Simulate H on input $\langle M, \langle M\rangle\rangle$
2. *Accept* if H rejects. *Reject* if H accepts."

D accepts $\langle M\rangle$ iff M doesn't accept $\langle M\rangle$.

D accepts $\langle D\rangle$ iff D doesn't accept $\langle D\rangle$.

Contradiction.

Why is this proof a diagonalization?

All TMs ⇩	All TM descriptions: $\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	. . .	$\langle D\rangle$
M_1	acc					
M_2		rej				
M_3			acc			
M_4				acc		
⋮						
D						

Recognizable vs Decidable

Recognizable L: we know when to halt when w is in L but not when w is not in L

Decidable L: we know when to halt either way

Intuitively, unbounded search space vs bounded search space.

E.g. A_{CFG} (search all possible derivations vs search all derivations of CNF of length $2|w| - 1$)

$\overline{A_{\text{TM}}}$ is T-unrecognizable

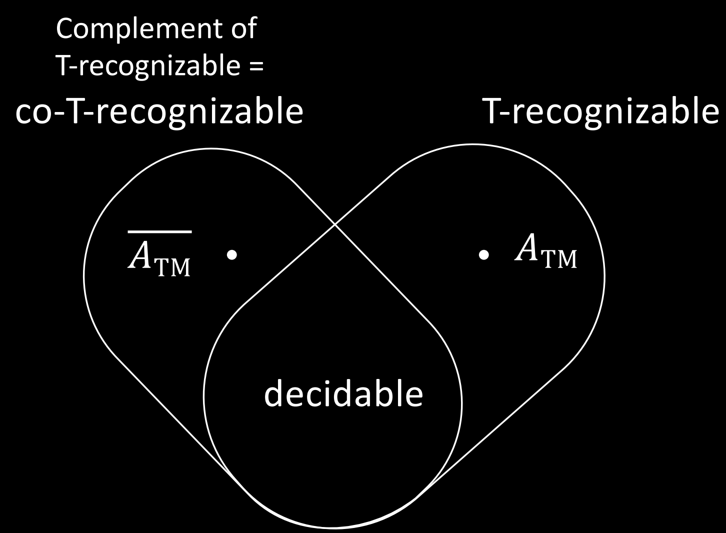
Theorem: If A and $\overline{A}$ are T-recognizable then A is decidable

Proof: Let TM M_1 and M_2 recognize A and $\overline{A}$.

Construct TM T deciding A.

T = "On input w

1. Run M_1 and M_2 on w in parallel until one accepts.
2. If M_1 accepts then *accept*.
   If M_2 accepts then *reject*."



Corollary: $\overline{A_{\text{TM}}}$ is T-unrecognizable

Proof: A_{TM} is T-recognizable but also undecidable

The Reducibility Method

Use our knowledge that A_{TM} is undecidable to show other problems are undecidable.

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

Theorem: $HALT_{\mathrm{TM}}$ is undecidable
Proof by contradiction, showing that A_{TM} is reducible to $HALT_{\mathrm{TM}}$:

Assume that $HALT_{\mathrm{TM}}$ is decidable and show that A_{TM} is decidable (false!).
Let TM R decide $HALT_{\mathrm{TM}}$.

Construct TM S deciding A_{TM}.

$S =$ "On input $\langle M, w\rangle$

1. Use R to test if M on w halts. If not, reject.
2. Simulate M on w until it halts (as guaranteed by R).
3. If M has accepted then *accept*.
   If M has rejected then *reject*.

TM S decides A_{TM}, a contradiction. Therefore $HALT_{\mathrm{TM}}$ is undecidable.

Quick review of today

- A_{TM} is undecidable
- The diagonalization method
- $\overline{A_{\text{TM}}}$ is T-unrecognizable
- The reducibility method
- Halting Problem