CSCI 361 Lecture 14: Undecidability

Shikha Singh

Announcements & Logistics

- Hand in **reading assignment # 9**
- Pick up **reading assignment #10**
  - Due start of class on Thur Oct 31
- **HW 5** due this Wed Oct 30

Last Time

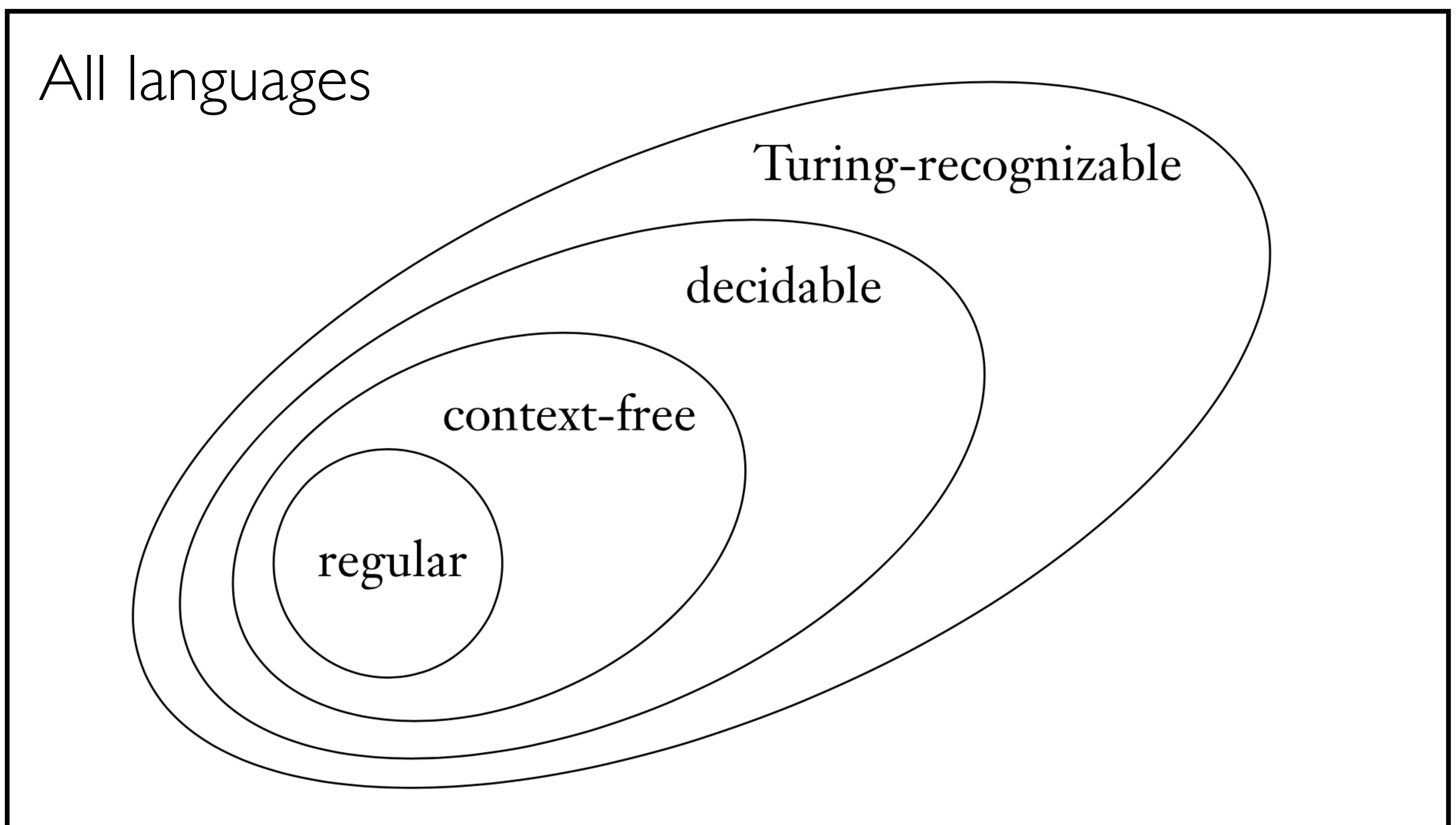
- Discussed many examples of decision problems that are decidable
  - $A_{\text{DFA}}, A_{\text{CFG}}$: Does a given DFA/CFG accept a given string
  - $E_{\text{DFA}}, E_{\text{CFG}}$: Is the language of the given DFA/CFG empty?
  - Other variations using reduction to the above
- All these problems are about ***semantic properties*** of DFA/CFG

Today

- Show that similar semantic properties of TMs are undecidable
- Develop a strategy for recognizing and proving a bunch of languages are undecidable by TMs

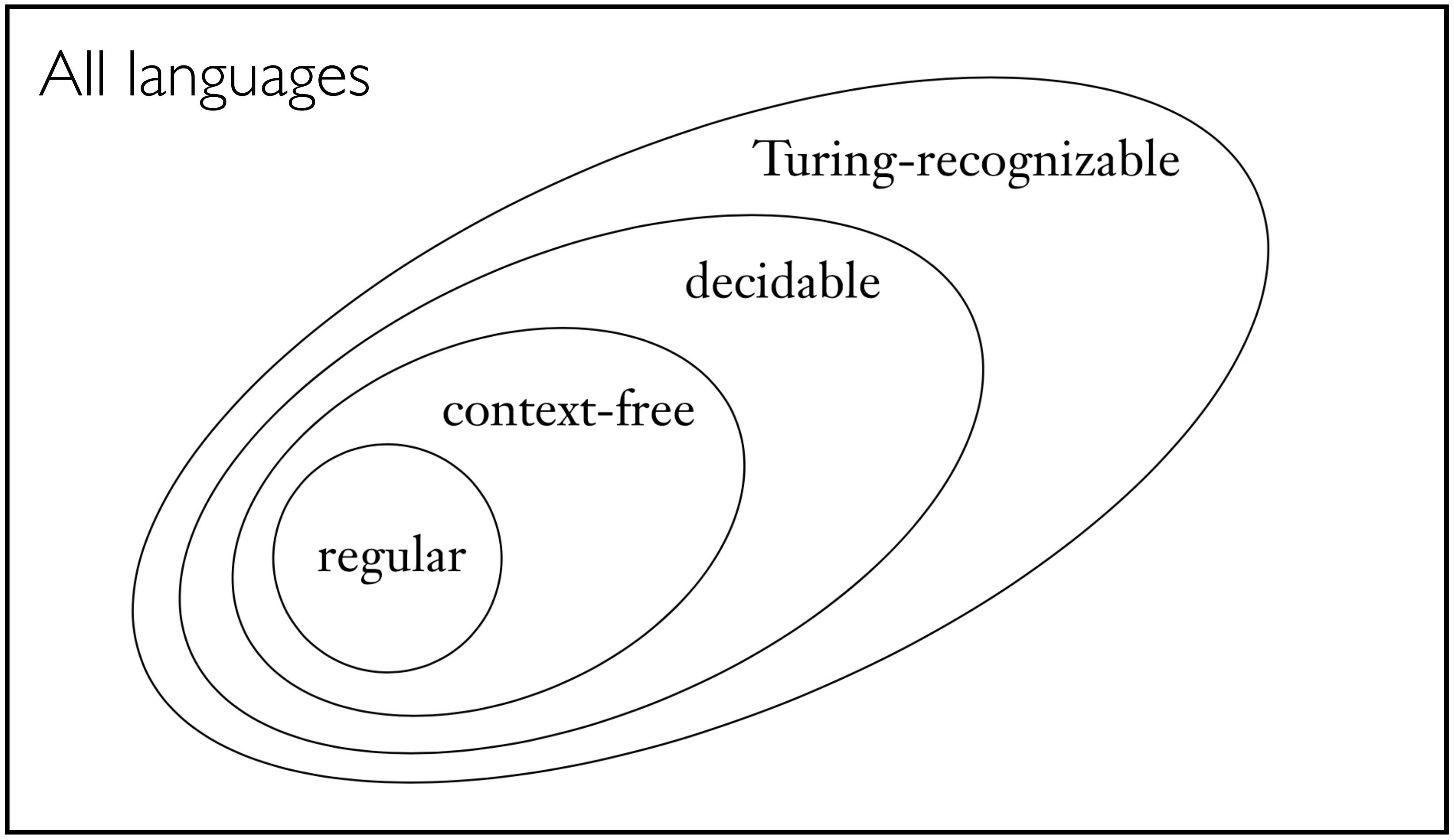
Countability Argument

- There are **many** languages that cannot even be recognized by TMs
- Can argue by comparing set of all TMs to set of all languages



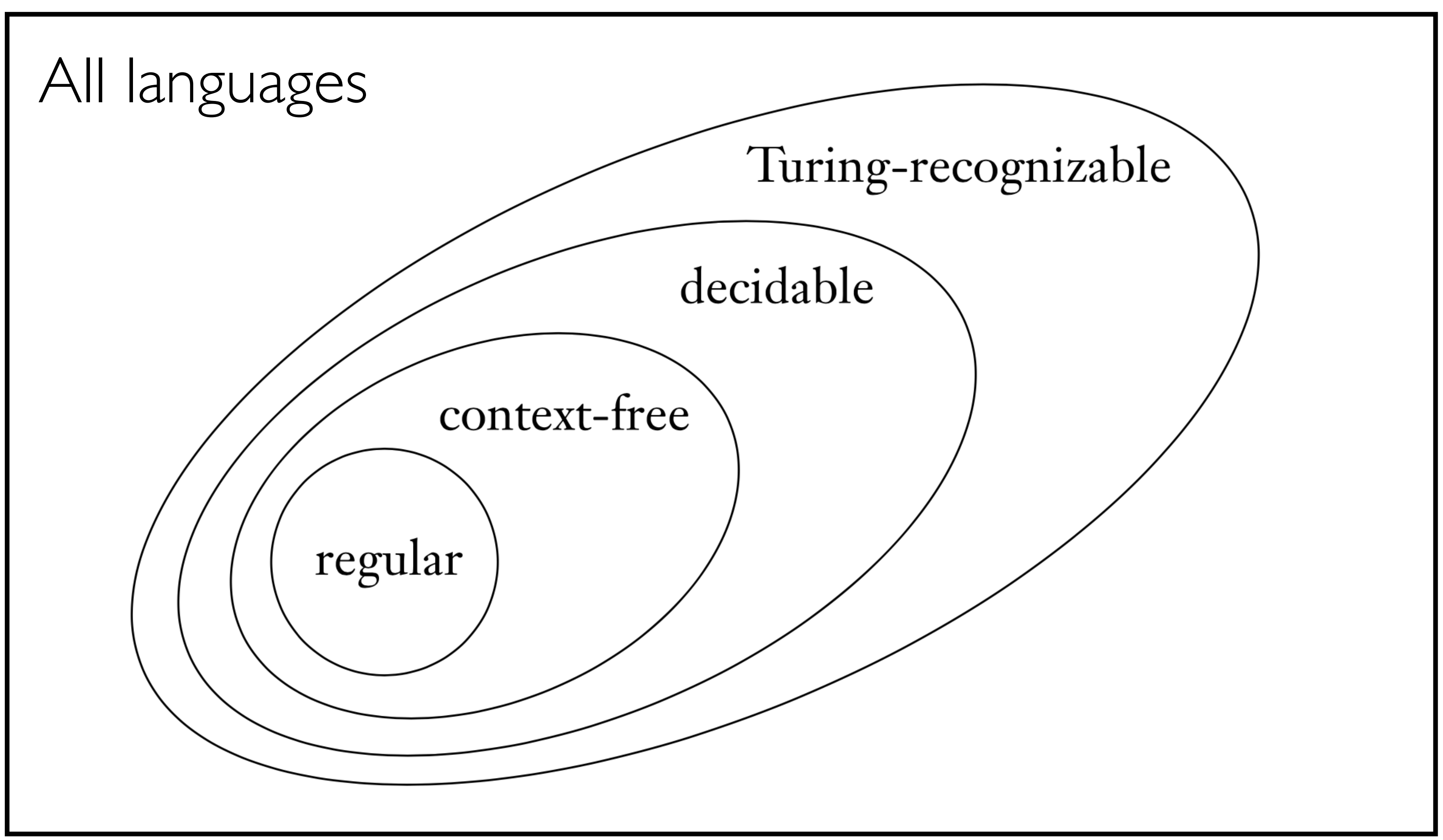
Countability Argument

- **Question.** Why is the set of all TM's countable?
  - Σ^* is countable for any finite alphabet
  - TM's can be encoded: $M \rightarrow \langle M \rangle$ over a finite Σ



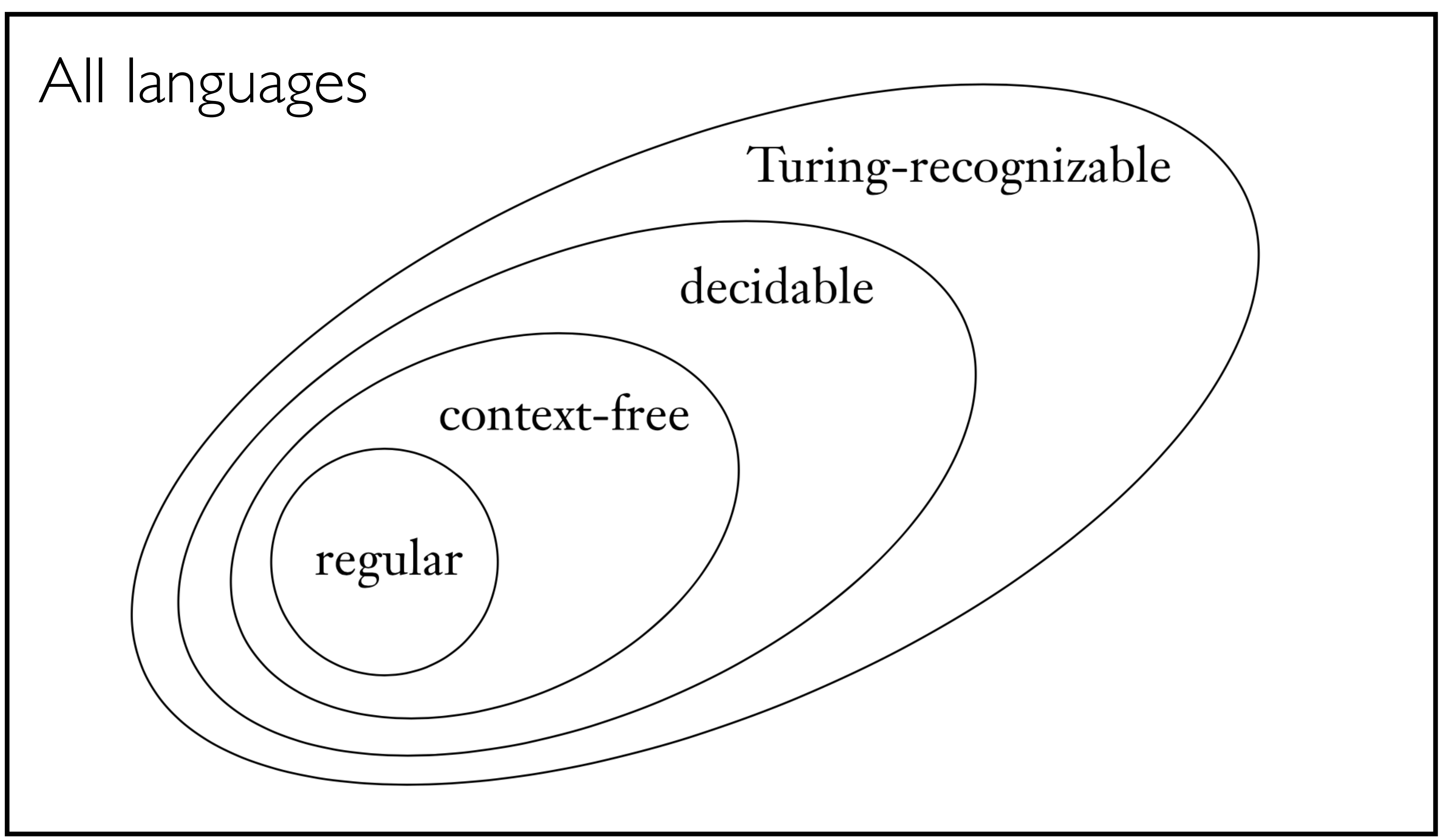
Countability Argument

- **Question.** Why is the **set of all languages** $\mathscr{L}$ uncountable?
  - Mapping between $\mathscr{L}$ and set of infinite binary sequences
  - Alternatively: $\mathscr{L}$ is the power set of Σ^*



Countability Argument

- **Takeaway**: There are infinitely many decision problems that cannot be solved by any TM
- **Today:** Specific problems that are undecidable and unrecognizable



Acceptance by TMs

- Consider the problem of given a TM and a string if the TM accepts the string, that is,
  $A_{\text{TM}} = \{\langle M, w \rangle \mid T \text{ is a TM and } w \in L(M)\}$
- Can we build a TM to decide this language?
  - Design a TM D such that D accepts $\langle M, w \rangle$ iff M accepts w
  - Such a TM is called a **universal TM** as it can simulate any TM

Universal Turing Machine

- Consider

  $$A_{\text{TM}} = \{\langle M, w \rangle \mid T \text{ is a TM and } w \in L(M)\}$$

- Let D be the following TM
  - On input $\langle M, w \rangle$
    - Run M on w, accept iff M accepts
- **Question.** Does D decide A_{TM}?
  - No! May loop forever on $\langle M, w \rangle$ if M loops forever on w
- **Question.** Does D recognize A_{TM}?
  - Yes! Thus A_{TM} is TM recognizable.

Theorem: A_{TM} is Undecidable

- **(Proof by contradiction.)** Suppose H is a decider for A_{TM}

$$H(\langle M, w\rangle) = \begin{cases} accept & \textbf{if } M \textbf{ accepts } w \\ reject & \textbf{if } M \textbf{ does not accept } w. \end{cases}$$

- Consider TM D that uses H as follows:
- $D =$ "On input $\langle M\rangle$, where M is a TM
  1. Run H on $\langle M, \langle M\rangle\rangle$
  2. If H accepts, then **reject**; If H rejects then **accept**
- **Question.** D takes as input a TM and is itself a TM, how can we get a contradiction?

Theorem: A_{TM} is Undecidable

- **(Proof by contradiction.)** Suppose H is a decider for A_{TM}

$$H(\langle M, w\rangle) = \begin{cases} accept & \textbf{if } M \textbf{ accepts } w \\ reject & \textbf{if } M \textbf{ does not accept } w. \end{cases}$$

- Consider TM D that uses H as follows:
- $D =$ "On input $\langle M\rangle$, where M is a TM
  1. Run H on $\langle M, \langle M\rangle\rangle$
  2. If H accepts, then reject; If H rejects then accept
- **Final step.** If we give D, the input $\langle D\rangle$, then
  - D accepts $\langle D\rangle$ iff D rejects $\langle D\rangle$ $\Rightarrow \Leftarrow$ ∎

Theorem: A_{TM} is Undecidable

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\cdots$
M_1	*accept*		*accept*		
M_2	*accept*	*accept*	*accept*	*accept*	
M_3					$\cdots$
M_4	*accept*	*accept*			
$\vdots$		$\vdots$			

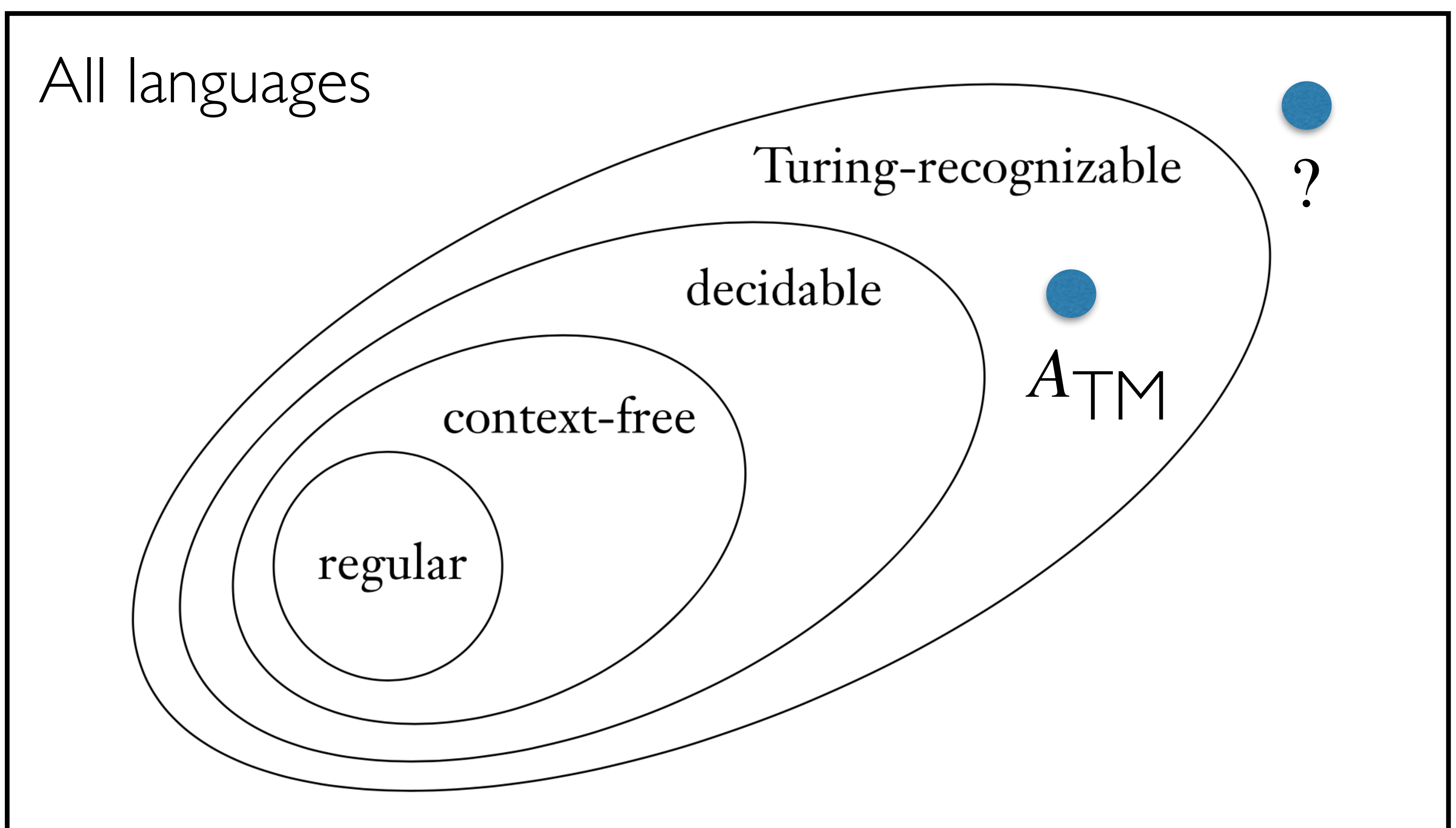
Entry i, j is *accept* if M_i accepts $\langle M_j \rangle$

Theorem: A_{TM} is Undecidable

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\cdots$	$\langle D \rangle$	$\cdots$
M_1	<u>*accept*</u>	*reject*	*accept*	*reject*		*accept*	
M_2	*accept*	<u>*accept*</u>	*accept*	*accept*	$\cdots$	*accept*	$\cdots$
M_3	*reject*	*reject*	<u>*reject*</u>	*reject*		*reject*	
M_4	*accept*	*accept*	*reject*	<u>*reject*</u>		*accept*	
$\vdots$		$\vdots$			$\ddots$		
D	*reject*	*reject*	*accept*	*accept*		<u>?</u>	
$\vdots$		$\vdots$					$\ddots$

TM Unrecognizable Language

- **Question.** A language that is neither decidable nor recognizable?
- **Lemma.** A language L is TM decidable iff both L and its complement $\overline{L}$ are Turing recognizable.



Turing Recognizable vs Decidable

Lemma. A language L is TM decidable iff both L and its complement $\overline{L}$ are Turing recognizable.

Proof.

($\Rightarrow$) By definition

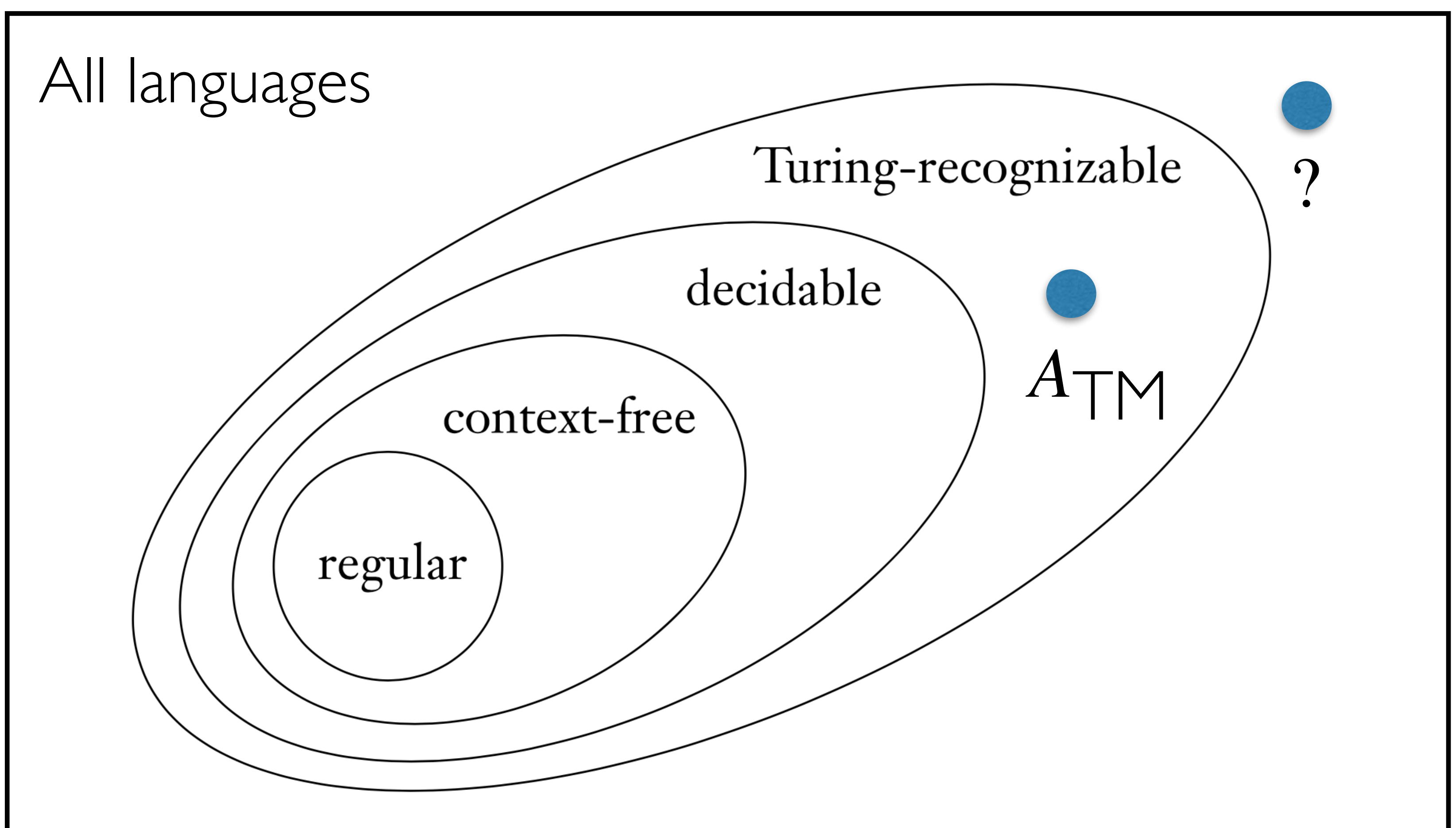
($\Leftarrow$) Consider M_L and $M_{\overline{L}}$ that recognize L and $\overline{L}$. Simulate both in parallel using two tapes

Accept if M_L accepts and reject if $M_{\overline{L}}$ accepts

To appreciate the distinction between TM decidable and TM recognizable, let's see an example of the latter.

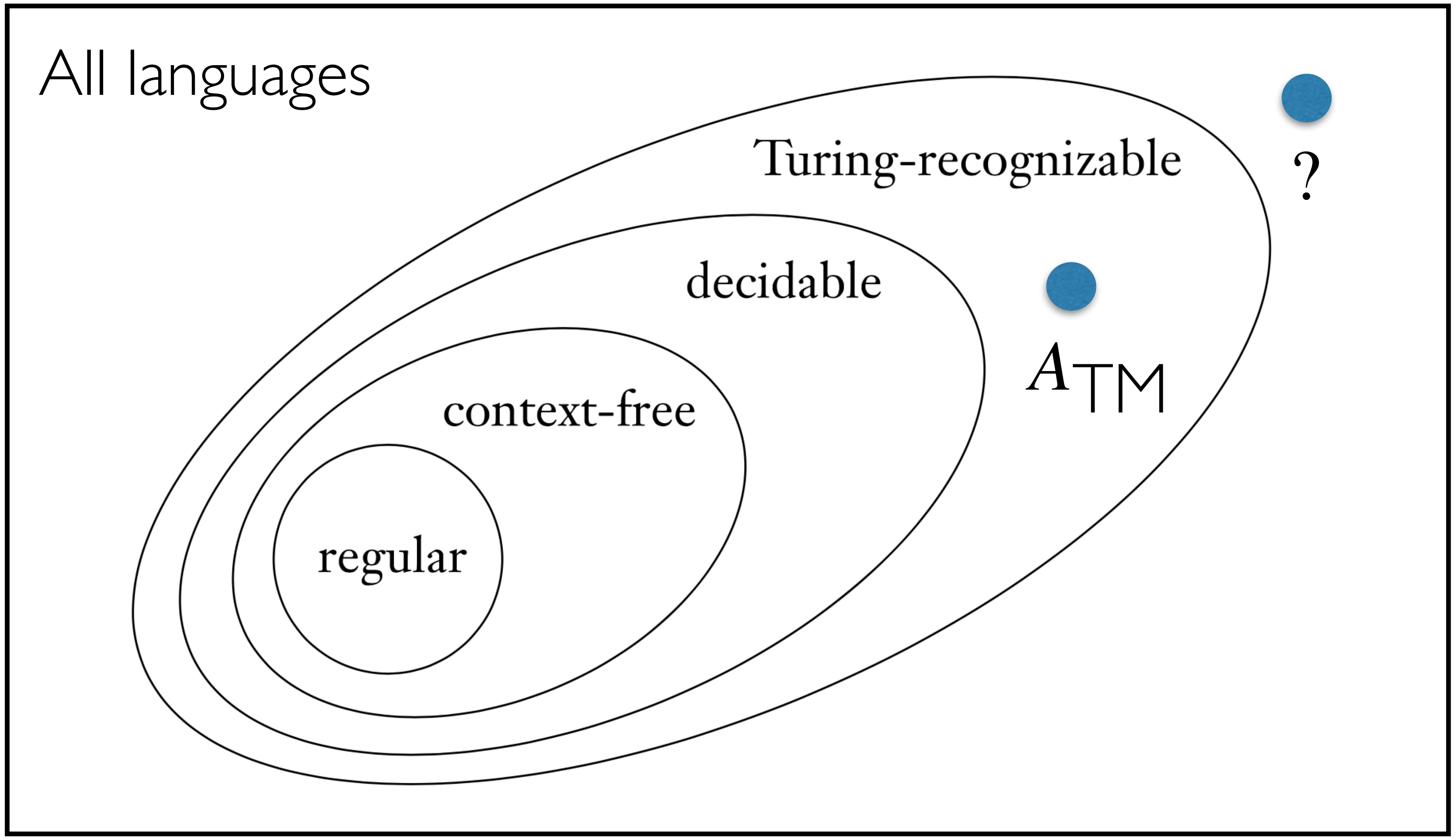
TM Unrecognizable Language

- **Lemma.** A language L is TM decidable iff both L and its complement $\overline{L}$ are Turing recognizable.
- **Question.** What language is **not** TM recognizable?



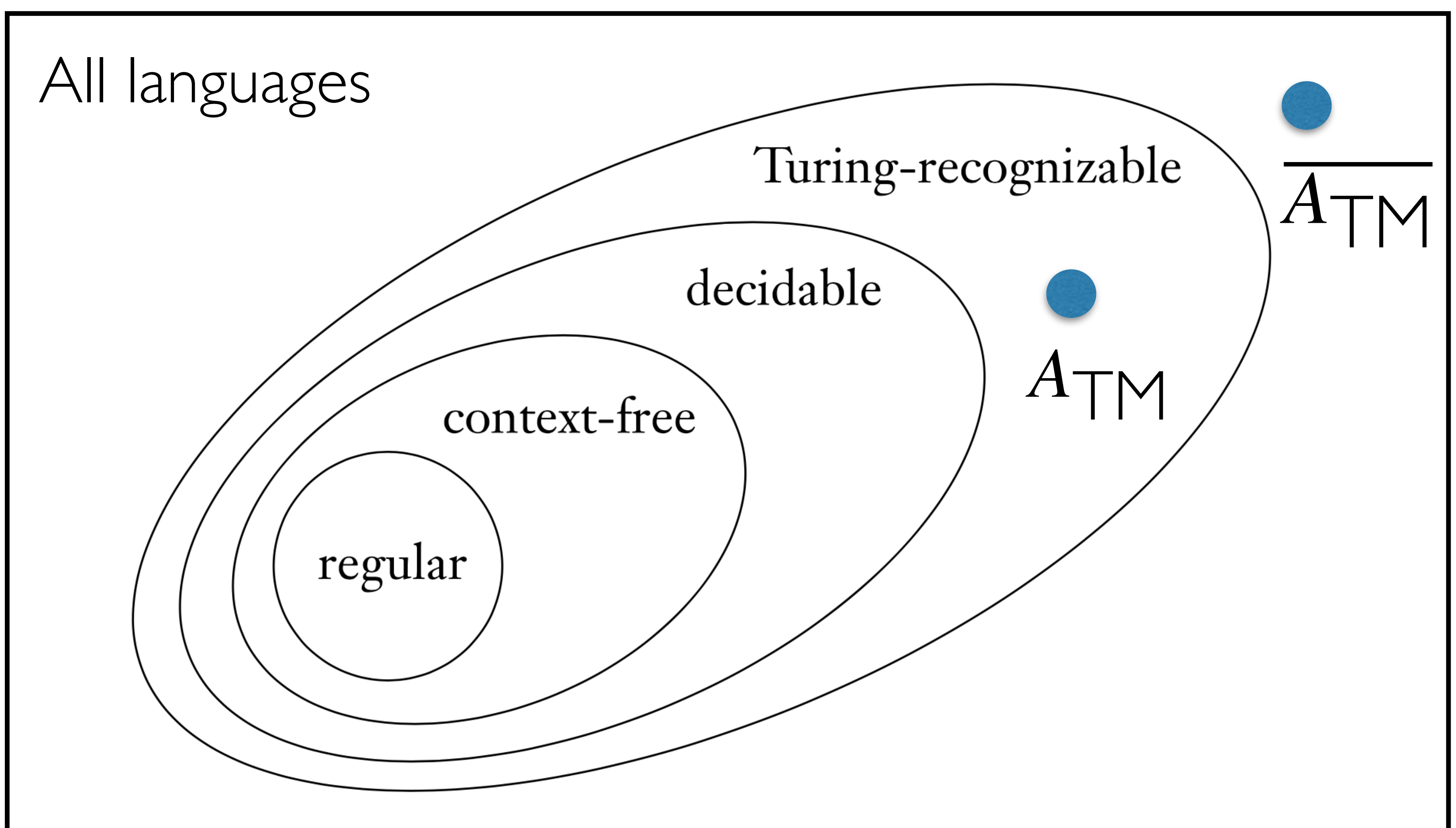
TM Unrecognizable Language

- **Lemma.** A language L is TM decidable iff both L and its complement $\overline{L}$ are Turing recognizable.
- **Question.** What language is **not** TM recognizable?
    - $\overline{A_{\text{TM}}}$



TM Unrecognizable Language

- **Lemma.** A language L is TM decidable iff both L and its complement $\overline{L}$ are Turing recognizable.
- **Question.** What language is **not** TM recognizable?
  - $\overline{A_{\mathsf{TM}}}$



Halting Problem

- More natural and practical computational problem:
  - Does a given program ever go into an infinite loop on some input?
- Given a TM M and string w, does M halt on w?
  - $\text{HALT}_{\text{TM}} = \{\langle M, w \rangle \mid M \text{ is a TM and } M \text{ halts on } w\}$
- How can we show that this problem is undecidable?

Reductions to Prove Undecidability

- **Problem.** Show that A is undecidable.
- **Reduction-based proof:**
  - Assume A is decidable
  - Reduce a **known undecidable problem** B to A (show that solving A would also solve B)
  - Reach a contradiction $\Longrightarrow A$ cannot be decidable
- Which problem to use to show halting problem is undecidable?
  - A_{TM} (the only problem so far we know that is undecidable)

Halting Problem Undecidability Proof

- **Theorem.** HALT_{TM} is undecidable.
- Proof Idea:
  - We know A_{TM} is undecidable
  - Want to show HALT_{TM} is undecidable
  - Need to reduce one to the other
  - Which direction does the reduction go?

Halting Problem Undecidability Proof

- **Theorem.** HALT_{TM} is undecidable.
- **Proof.** Suppose TM R decides HALT_{TM}.
- Construct a decider S for A_{TM}:
- $S =$ " On input $\langle M, w \rangle$,
  1. Run R on $\langle M, w \rangle$.
  2. If R rejects, then reject.
  3. If R accepts, then simulate M on w. If M enters accept state, then accept; if M enters reject state, then reject.
- S decides A_{TM} but A_{TM} is undecidable. $\Rightarrow \Leftarrow$ ∎

Does M Accept Anything?

- **Theorem.**
  $E_{\mathrm{TM}} = \{\langle M \rangle \mid M$ is a TM and $L(M) = \varnothing\}$ is undecidable.
- **Proof Idea.** Suppose TM R decides E_{TM}
- **Question.** Can R be used to decide A_{TM}?
  - Input to A_{TM} is $\langle M, w \rangle$ and input to R is just a TM
  - Want a TM M_w such that determining if $L(M_w)$ is empty or not determines whether w is accepted by M or not

E_{TM} is Undecidable

Proof. Suppose TM R decides E_{TM}. Consider the following decider D for A_{TM}:

- $D =$ "On input $\langle M, w \rangle$
  - Encode a TM M_w that does the following:
    - $M_w =$ "On input x,
      - If $x \neq w$, reject.
      - If $x = w$, then run M on w and accept if M does, else reject.
  - **Question.** What can say about $L(M_w)$?
  - Run R on $\langle M_w \rangle$. If R accepts, reject; if R rejects, accept.

w is hardcoded in description of M_w, and not part of input of M_w

Exercise

Problem. Show that

$EQ_{\mathsf{TM}} = \{\langle M, N \rangle \mid M, N$ are TMs and $L(M) = L(N)\}$ is undecidable.

Hint. Reduce E_{TM} to it.