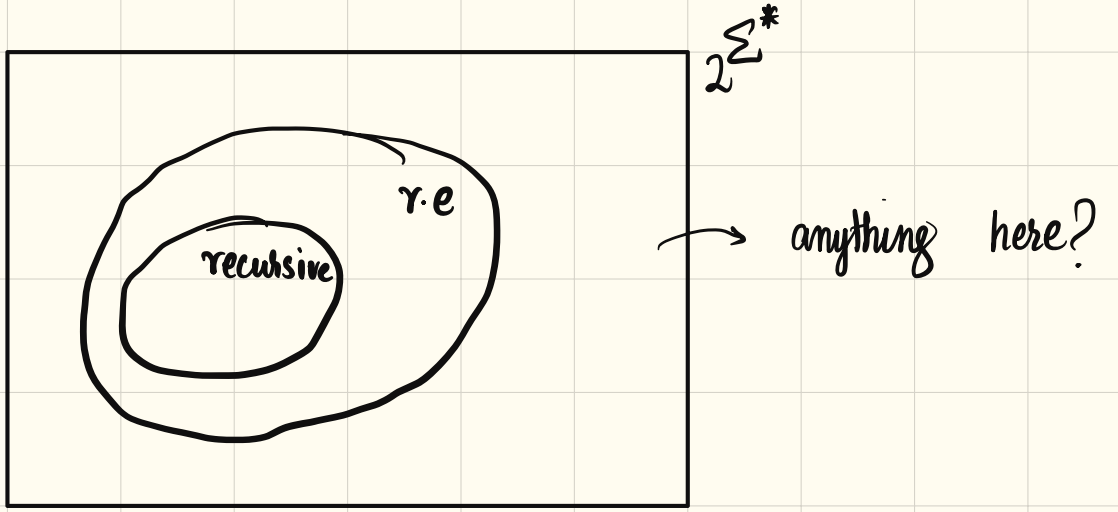
02 Apr 2025 – Theory of Computation – Week 12

$2^{\mathbb{N}}$ = set of subsets of $\mathbb{N}$

Lemma: $2^{\mathbb{N}}$ is uncountable (diagonalization)



Trying to prove: $\exists\ L \subseteq \Sigma^*$ s.t. L is not r.e.

TM := set of all turing machines → valid encodings of

→ for one TM, if Σ is fixed, $|Q|, |\Sigma|, |\Gamma|$

$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$

Finite size to describe TM

Is TM countable ?

OR

Is there a way to enumerate all turing machines?

Claim: TM is countable.

For every TM

$(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$

⇓

encode as a binary string

$\Sigma_u = \{0, 1\}$

e.g.:

$$\langle 1^{|Q|} 0 \; 1^{|\Sigma|} 0 \; 1^{|\Gamma|} 0 \; \ldots$$

states $1 \rightarrow |Q|$

alphabet $1 \rightarrow |\Sigma|$

$$(q, a) \longrightarrow (q', b, L)$$

$$1^q 0 \; 1^a 0 \; 1^{q'} 0 \; 1^b 0$$

Every TM $\longrightarrow$ has an encoding $\longrightarrow$ may have multiple encodings if order for describing δ is not decided

Not all strings correspond to a TM.

Set of all languages, $\mathcal{L} = 2^{\Sigma^*}$ over Σ^*

$= \{ L \mid L \subseteq \Sigma^* \}$

$\mathcal{L} \longrightarrow$ uncountable
$TM \longrightarrow$ countable
$\Big\} \longrightarrow$ There is some language not recognizable by a TM.

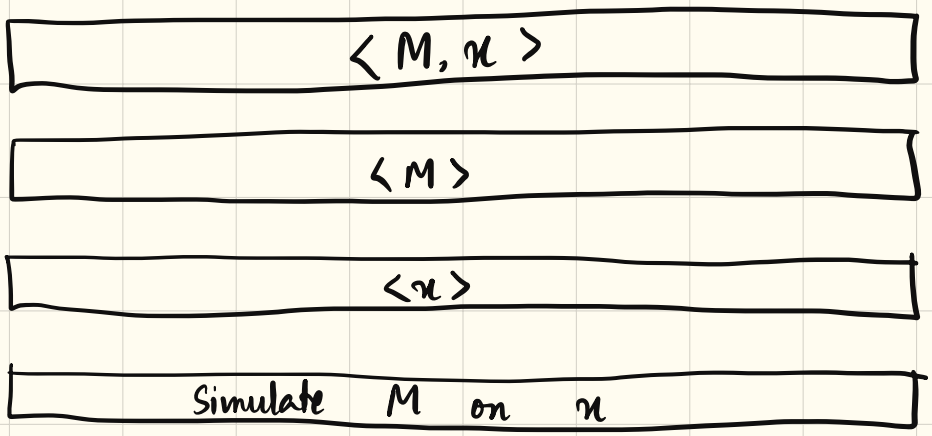
Given: $\langle M \rangle$ an encoding of a TM.
and an input x for M.

Output: Does M accept x?

Construction of M' : input = a turing machine and an input string

$= \langle M, x \rangle$

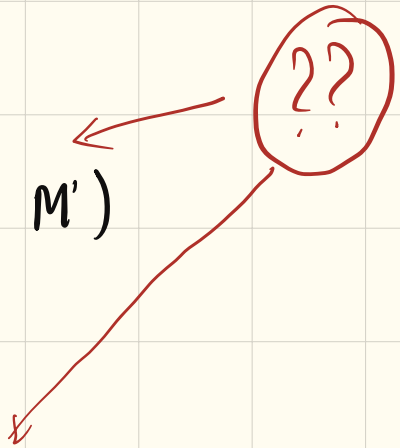
1. Simulate M on x.
2. Accept / Reject if M accepts / rejects x.



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

↓

language (membership problem)



MP is r.e (by virtue of M')

Q: Is MP decidable / recursive?

No → prove by contradiction.

Proof: Suppose MP is decidable $\Rightarrow \exists$ a halting TM T_1 s.t. M decides MP.

T_1 on input $\langle M, x \rangle$ = $\begin{cases} \text{accepts} & \text{if } M \text{ accepts } x \\ \text{rejects} & \text{o/w} \end{cases}$

always halts

Construct another TM D.

On input $\langle M \rangle$

1) Run $T_1(\langle M \rangle, \langle M \rangle)$ → read both as encoding and as a string.
2) Complement the answer.

Claim

D accepts an input $\langle M \rangle$ iff. T_1 rejects $(\langle M \rangle, \langle M \rangle)$

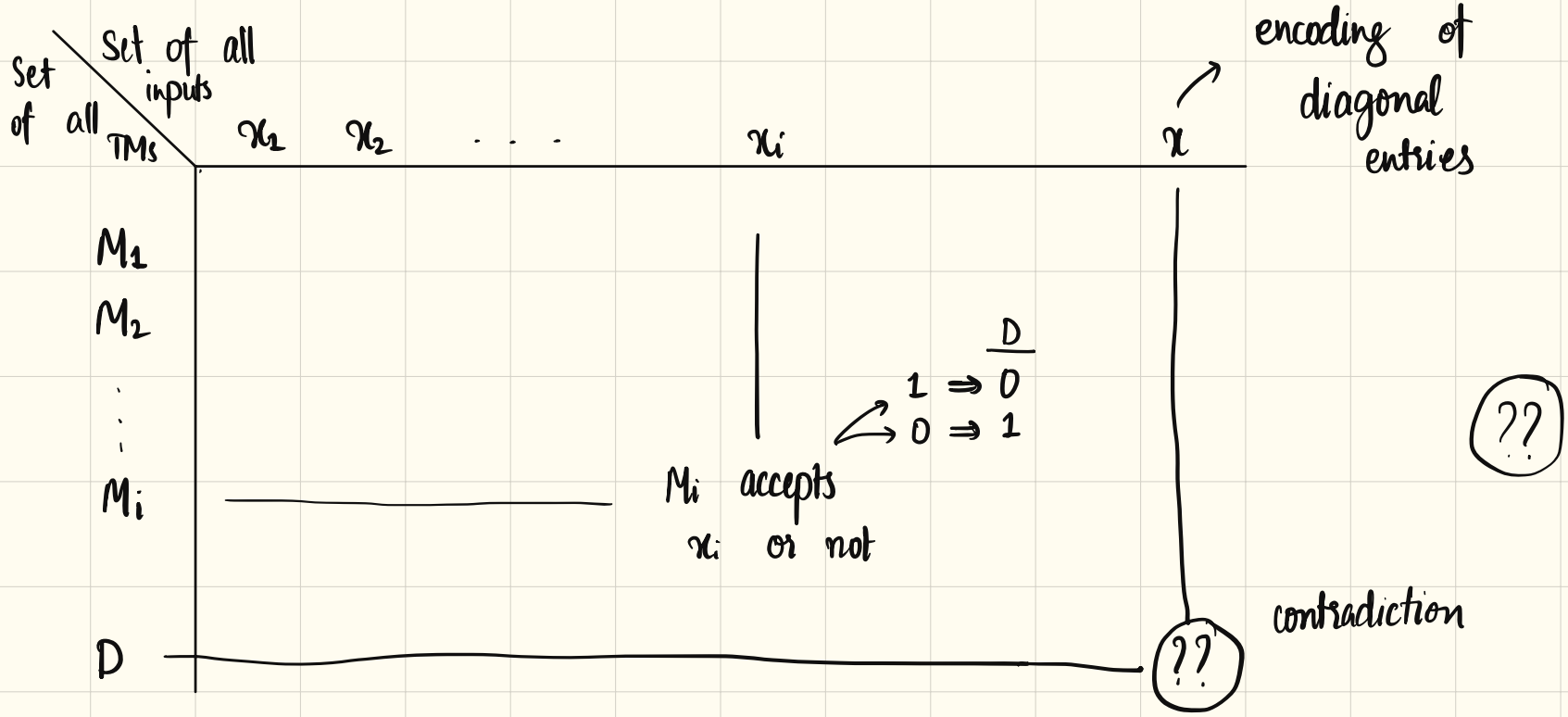
iff M does not accept $\langle M \rangle$.

Q: What happens if input to D is $\langle D \rangle$?

D accepts $\langle D \rangle$ iff D does not accept $\langle D \rangle$

$\Rightarrow \Leftarrow$

Equivalent proof by diagonalization



Like sets ⇝ (but) changing TM def^{n} does not make sense

definition problem

??