Not All Languages are Recognizable

TMCODE $= \{x \mid x \in \{0,1\}^*$ is a valid encoding of a $1-$tape Turing machine$\}$.

$\mathcal{L}$: the set of all *languages* over Σ.

TMCODE is countably infinite (subset of a countably infinite set)

but

$\mathcal{L}$ is uncountably infinite.

Universal Turing Machines

A TM U with input:

description $e(M)$ of a 1-tape DTM $M = (Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r)$

description $e(w)$ of a string $w \in \Sigma^*$

U **simulates** M and accepts $e(M)e(w)$ iff M accepts its input w.

Universal Turing Machines

A universal Turing machine U takes as input $e(M)e(w)$.

M is a 1-tape DTM and w is a string over M's input alphabet.

U simulates M on input w using three tapes:

Tape 1: $e(M)e(w)$.
Tape 2: Contents of the tape of M.
Tape 3: Current state of M.

Copy $e(w)$ from Tape 1 to Tape 2;

Write e(start state) of M on Tape 3;

Simulate M:

Use the current state of M (Tape 3) and the current symbol (Tape 2) to find the next state q, symbol a, and direction D from Tape 1;

Write the next state q on Tape 3;

Write the symbol a on Tape 2;

Move in the direction D on Tape 2;

Halt and accept if M accepts;

Halt and reject if M rejects;

An Undecidable Language

$$A_{\mathrm{TM}} = \{e(M)e(w) \mid M \text{ is a Turing machine that accepts } w\}.$$

The *Acceptance* problem for TMs.

(Text calls it the Halting problem.)

A_{TM} is recognizable.

An Undecidable Language

A_{TM} is undecidable.

Suppose not. Let H be a decider for A_{TM}.

H on input $e_H(M, e_M(w))$:

Accepts $e_H(M, e_M(w))$ if M accepts w .

Rejects $e_H(M, e_M(w))$ if M does not accept w.

Construct a TM D that on input $e_D(M)$ (M is a TM) functions as follows:

Run H on input $e_H(M, e_M(M))$.

Accept $e_D(M)$ if H rejects $e_H(M, e_M(M))$.

Reject $e_D(M)$ if H accepts $e_H(M, e_M(M))$.

When does D accept $e_D(M)$?

If H rejects $e_H(M, e_M(M))$ - that is if M does not accept $e_M(M)$.

When does D reject $e_D(M)$?

If H accepts $e_H(M, e_M(M))$ - that is if M accepts $e_M(M)$.

An Undecidable Language

The decider D exists since we assumed H exists.

Now, run D on $e_D(D)$.

D accepts $e_D(D)$ if D does not accept $e_D(D)$.

D rejects $e_D(D)$ if D accepts $e_D(D)$.

Contradiction. Therefore, our assumption is wrong: H doesn't exist.

A_{TM} is undecidable.

When is a Recognizable Language Decidable?

If and only if the language and its complement are both recognizable.

One direction follows:

L is decidable $\Rightarrow$ $\overline{L}$ is decidable.

A decidable language is recognizable.

Suppose L and $\overline{L}$ are recognizable. To show that they are decidable.

Let M_1 accept L and M_2 accept $\overline{L}$.

A decider for L runs M_1 and M_2 in parallel.

On input w:

1. $K \leftarrow 1$;
2. Run M_1 on w for K steps.
3. If M_1 accepts then halt and accept.
4. If M_1 rejects then halt and reject.
5. Run M_2 on w for K steps.
6. If M_2 accepts then halt and reject.
7. If M_2 rejects then halt and accept.
8. Increment K and repeat from step 2.

A decider because M_1 or M_2 will halt on w.

A Language that is not Recognizable

The complement $\overline{A_{\text{TM}}}$ of A_{TM}.