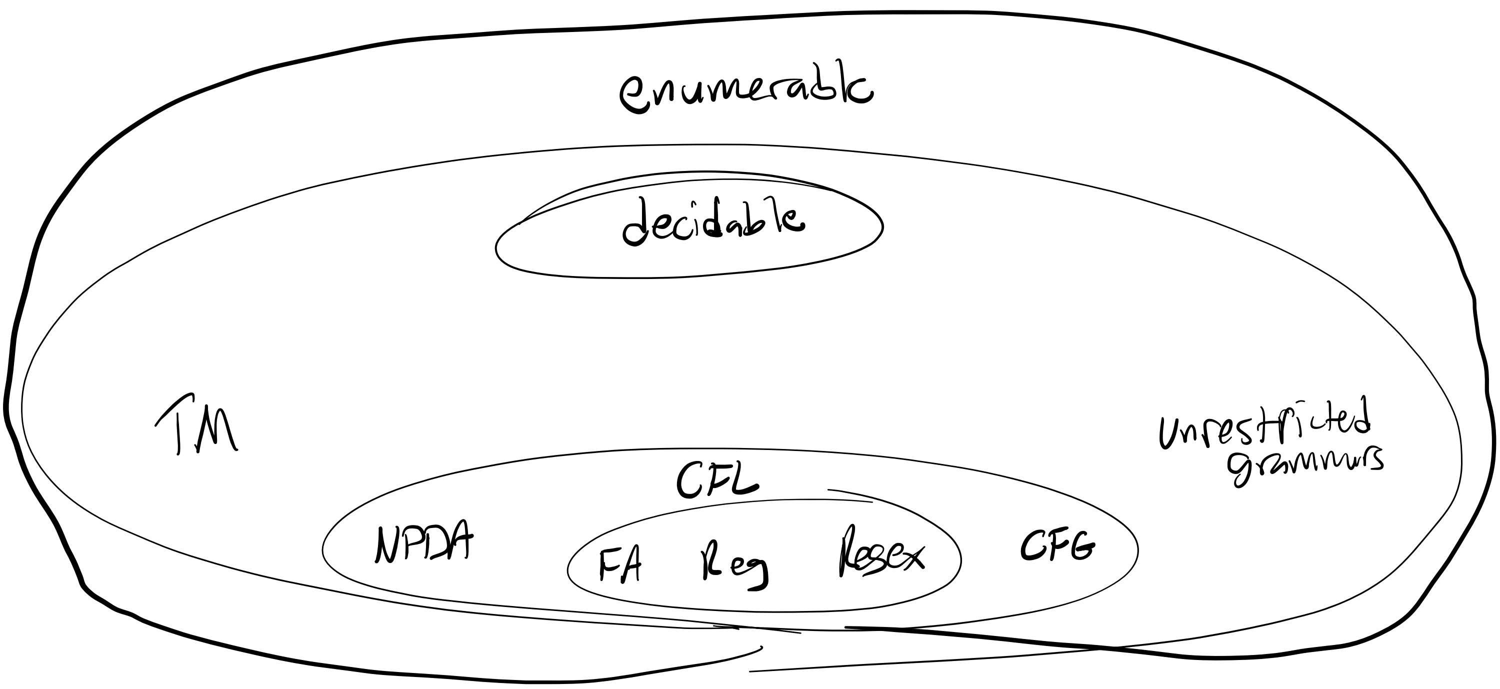
CSCI 301 - Lecture 35: Church-Turing Thesis
Computability

The Church-Turing Thesis

Any model of algorithmic computation is no more powerful than a Turing Machine.



TM outcomes : M w

- M accepts w
- M rejects w
- M does not terminate

A language L is decidable if there exists a machine M where $L(M) = L$ and M terminates on all inputs.

$$L = \{(M, w) : M \text{ accepts } w\}$$

M': Simulate M on w
if M accepts accept
M rejects reject

$L_{DFA} = \{\langle M_{DFA}, w\rangle : M \text{ accepts } w\}$ decidable

$L_{NFA} = \{\langle M_{NFA}, w\rangle : M \text{ accepts } w\}$ decidable

L_{PDA} ? $\equiv L_{CFG} \equiv L_{CNF} \rightarrow$ finite steps, decidable!

L_{TM} not decidable

Thm: HALT is not decidable

Proof: Suppose $H(\langle M, x \rangle)$ exists

Construct $Z(\langle M, w \rangle)$ that:

1. Simulates H on $\langle M, w \rangle$
   - Loop forever if $H(\langle M, w \rangle)$ accepts
   - Accept if $H(\langle M, w \rangle)$ rejects

Run $Z(\langle Z, Z \rangle)$

If Z accepts, then H said Z didn't terminate

doesn't terminate, H said it did!

Reduction

Reduce Halt to A_{TM}

Suppose A_{TM} is decidable; so M_{TM} exists.

Construct $M_H(\langle M, x \rangle)$:

- Simulate M_{TM} on $\langle M, x \rangle$
  - if M_{TM} accepts, accept
- Otherwise construct M' to mirror M, except $q_{accept} \leftrightarrow q_{reject}$

Run M_{TM} on $\langle M', x \rangle$

if M_{TM} accepts, then M halted and rejected x; accepts

otherwise reject