Some Decidable Problems

Recall A language L is decidable if there is a TM M that halts on every input $x \in \Sigma^*$ and

$x \in L \Rightarrow M$ accepts x.

$x \notin L \Rightarrow M$ rejects x.

We show that several problems concerning finite automata and c.f. grammars are decidable.

Theorem Acceptance problem for DFAs.

$$A_{DFA} = \{ \langle D, w \rangle \mid \begin{array}{l} D \text{ is a DFA that} \\ \text{accepts input } w \end{array} \}$$

is decidable.

Proof Following TM decides A_{DFA}.

$M :=$ "On input $\langle D, w \rangle$,

Simulate D on w.

If D accepts, accept.

If D rejects, reject."

Note that input for M is $\langle D, w \rangle$.

input for D is w.

Clearly,

M accepts $\langle D, w \rangle \iff$ D accepts w.

Moreover, since D is a DFA, it reaches its accept or reject decision in n steps where $|w| = n$. Hence M is guaranteed to halt on every input $\langle D, w \rangle$. ▨

Note The above holds even if D were an NFA or a regular expression since the TM can always first convert into an equivalent DFA. One notes here that the procedures to convert NFA or regular expression into DFA are constructive (mechanical) and hence are implementable (algorithmic) on a TM.

Theorem (Non-) Emptiness Problem for DFAs.

$$E_{DFA} = \{ \langle D \rangle \mid D \text{ is a DFA s.t. } L(D) \neq \phi \}$$

is decidable.

Proof Note that $L(D) \neq \phi$ (i.e. D accepts at least one input string) iff in D, there is at least one accept state that is reachable from its start state. Hence a TM can decide whether $L(D) \neq \phi$ by employing a standard graph reachability algorithm on directed graphs. ▨

Theorem All- Problem for DFAs.

$$All_{DFA} = \{ \langle D \rangle \mid D \text{ is a DFA s.t. } L(D) = \Sigma^* \}$$

is decidable.

Proof Note that $L(D) \neq \Sigma^*$ (i.e. D rejects at least one input string) iff in D,

there is at least one non-accept state that is reachable from its start state. This can be decided as before. ▨

Theorem Equivalence Problem for DFAs.

$$EQ_{DFA} = \{ \langle D, R \rangle \mid D, R \text{ are DFAs s.t. } L(D) = L(R) \}.$$

is decidable.

Proof $L(D)$, $L(R)$ are sets of strings. We note the set-theoretic fact that

$$L(D) = L(R) \iff L(D) \setminus L(R) = \phi \text{ and } L(R) \setminus L(D) = \phi$$

$$\iff \left(L(D) \cap \overline{L(R)}\right) \cup \left(L(R) \cap \overline{L(D)}\right) = \phi$$

We also note that the class of regular languages is closed under complements, intersection and union. Hence it is possible to algorithmically construct a DFA M s.t.

$$L(M) = \left(L(D) \cap \overline{L(R)}\right) \cup \left(L(R) \cap \overline{L(D)}\right).$$

As observed, $L(D) = L(R) \iff L(M) = \phi$.

One can decide whether $L(M) = \phi$ using the TM that decides E_{DFA}, the Emptiness Problem for DFAs!

Thus the following TM M^* decides EQ_{DFA}.

M^* = "On input $\langle D, R \rangle$,

Constnut a DFA M s.t.

$L(M) = (L(D) \cap \overline{L(R)}) \cup (L(R) \cap \overline{L(D)})$.

Decide whether $L(M) = \phi$.

If $L(M) = \phi$, accept.

If $L(M) \neq \phi$, reject."

Theorem Acceptance problem for contex free Grammars.

$$A_{CFG} = \{ \langle G, w \rangle \mid G \text{ is a c.f. grammar that generates } w \}$$

is decidable.

Proof To decide whether G generates w, one can in principle try out all possible

derivations starting from the start variable S of the grammar. However since this procedure must halt, one can only try out finitely many derivations. This would work if depending on the length $|w| = n$, there was an a priori upper bound on the length of the derivation. We recall that for a grammar in Chomsky Normal Form (where rules are of the form $A \to BC$, $A \to a$), a string w, $|w| = n$, if derivable, is derived in at most $2n$ steps. Moreover, one can algorithmically convert any grammar into an equivalent grammar in Chomsky Normal Form. Thus the following TM decides A_{CFG}.

M := " on input $\langle G, w \rangle$

Convert G into an equivalent grammar G' in Chomsky Normal Form.

Let $n = |w|$.

Try all derivations of G' with at

most $2n$ steps.

If any of these derivations derives w, then accept.

Else reject. "

Note The proof above shows that every Context Free language is decidable. If L is c.f. then it has underlying grammar G that can, w.l.o.g., be assumed to be in the Chomsky Normal form.

To decide if $w \in L$, one tries out all possible derivations of $\leq 2n$ steps where $|w| = n$.

This "algorithm" is inenfficient and takes time $2^{O(n)}$. It is possible to design $O(n^3)$ time algorithm using "dynamic programming" as we will see later.

Theorem (Non-) Emptiness Problem for c.f. grammars.

$$E_{CFG} = \{ \langle G \rangle \mid G \text{ is a c.f. grammar s.t. } L(G) \neq \phi \}$$

is decidable.

Proof 1 We can examine the proof of the pumping lemma for c.f. languages and deduce that if a grammar G generates at least one string, then it generates a string of length at most $p = b^{n+2}$.

Here n = number of variables in G

b = max length of right hand side of any rule in G.

Indeed, p serves as the pumping length and if there is a string $s \in L(G)$, $|s| \geq p$, s can always be "pumped down". This shows that if $L(G) \neq \phi$, then $L(G)$ must contain a string of length at most p. Thus one decides whether $L(G) \neq \phi$ by trying out all strings (of terminals) of length at most p

and checking whether any of them is derived by the grammar. ▨

Proof 2 We can design an algorithm that marks (identifies) all the variables in G that derive a string of terminals (denoted Σ).

To begin with, every variable A for which there is a rule $A \to w$, $w \in \Sigma^*$, is marked.

Then we repeatedly mark every variable B such that there is a rule $B \to w_1 w_2 \cdots w_k$ where each w_i is either a terminal or a variable that has already been marked.

The procedure halts when no more variables are getting marked.

Clearly, $L(G) \neq \phi$ iff S (start variable) derives a string of terminals

iff S is marked when the procedure halts. ▨

Note It turns out that the All-problem and the Equivalence Problem for C.F. Grammars are undecidable!

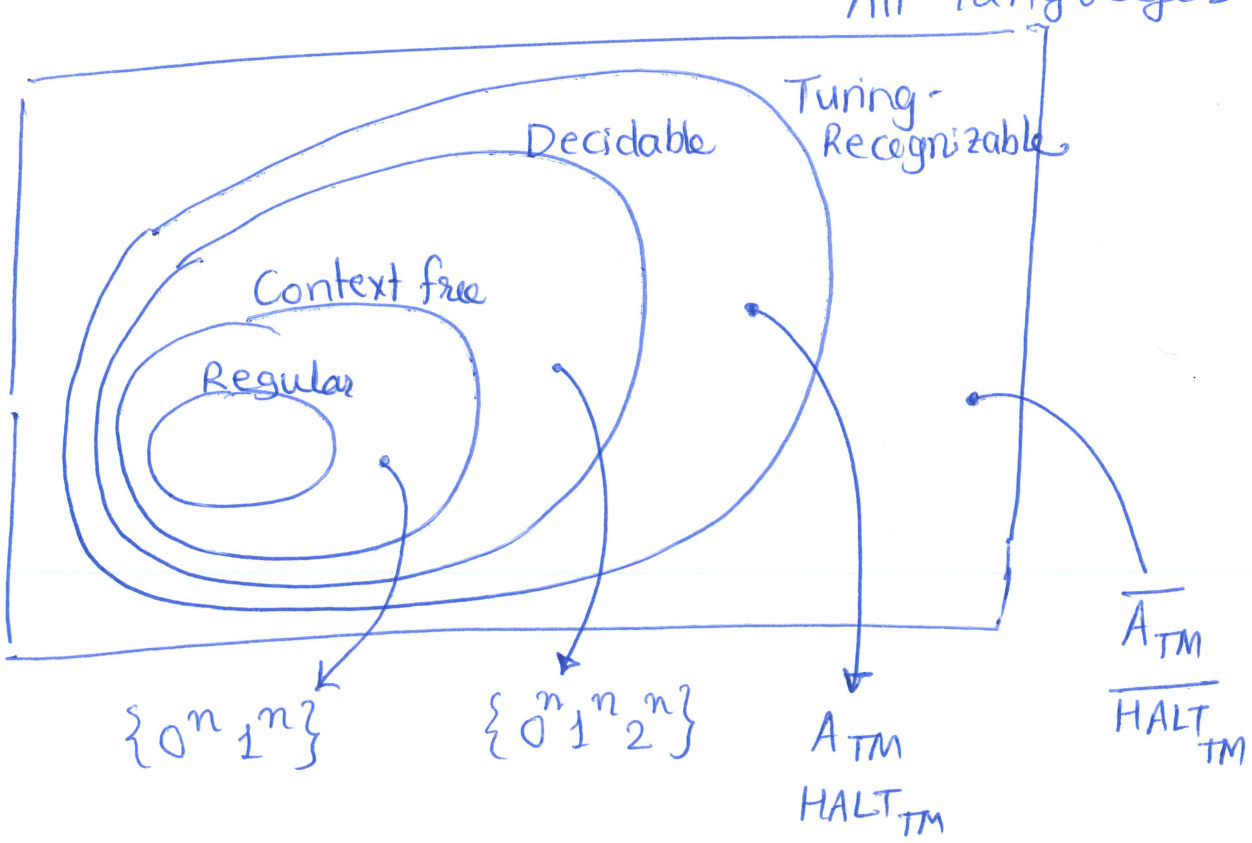
$$All_{CFG} = \{\langle G\rangle \mid G \text{ is a c.f.g. s.t. } L(G) = \Sigma^*\}.$$

$$EQ_{CFG} = \{\langle G, G'\rangle \mid \begin{array}{l} G, G' \text{ are c.f. grammars} \\ \text{s.t. } L(G) = L(G')\end{array}\}.$$

We will prove their undecidability if time permits.

— x —

To recall:



A_{TM} is the Acceptance Problem for TMs.

$HALT_{TM}$ " Halting "

$A_{TM} = \{ \langle M, w \rangle \mid$ M is a TM (description), $w \in \Sigma^*$ is input for M, M accepts w. $\}$

$HALT_{TM} = \{ \langle M, w \rangle \mid$ M is a TM that halts on input w. $\}$

We will now show that A_{TM}, $HALT_{TM}$ and many related problems are undecidable. The proof uses a technique called "the diagonalization method". The technique was first used by Cantor to prove that the set of real numbers is uncountable.

Def A set S is called countable if there exists an ordering of its elements

$$S = \{ s_1, s_2, s_3, s_4, s_5, \ldots \}.$$

The set could be finite or infinite (refered to as countably infinite in that case).

Note ① One may insist, w.l.o.g., that in the ordering each element $s \in S$ occurs exactly once.

② The definition only requires that the

ordering exists. When S is countably infinite, an equivalent definition would be that there exists a bijective (i.e. 1-to-1 and onto) map $f: \mathbb{N} \to S$ $\quad \mathbb{N} = \{1, 2, 3, 4, 5, \cdots\}$.

Examples

Fact Set of integers $\mathbb{Z}$ is countable. An ordering is

$$\mathbb{Z} = \{0, 1, -1, 2, -2, 3, -3, \cdots\}$$

Fact Set of rationals $\mathbb{Q}$ is countable. Note

$$\mathbb{Q} = \left\{ \frac{p}{q} \mid p, q \in \mathbb{Z}, \quad q \geq 1 \right\}.$$

It is tricky to exhibit an ordering here. One cannot order according to the magnitude ($|\frac{p}{q}|$), as is the case for $\mathbb{Z}$ above, since between every two rationals $0 < a < b$, there is a rational c s.t. $a < c < b$. However one can order all positive rationals $\frac{p}{q}$ according to increasing value of $p+q$. !

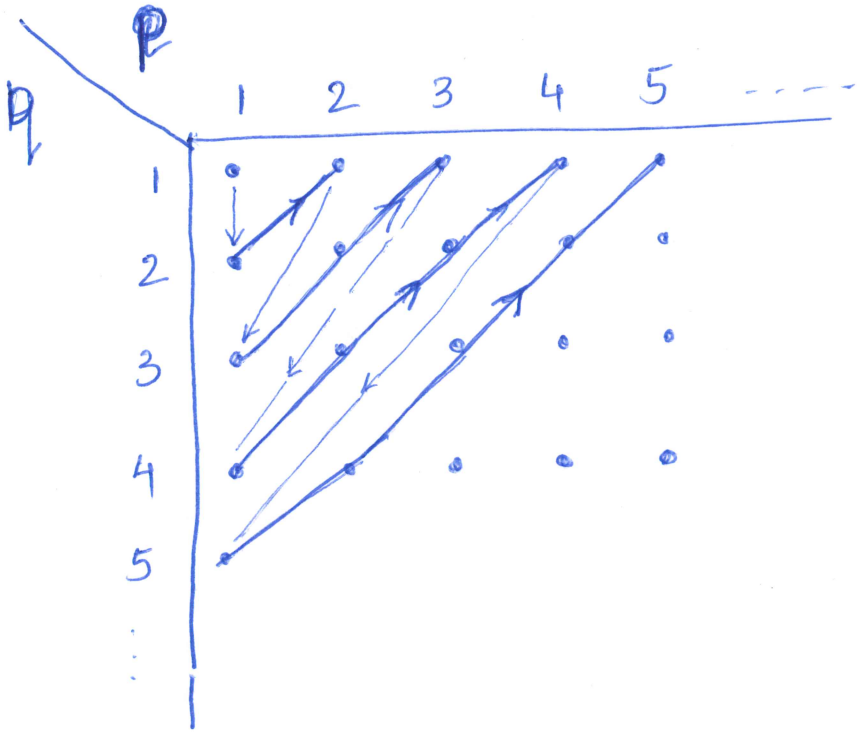
$$\mathbb{Q}^+ = \mathbb{Q}_2 \cup \mathbb{Q}_3 \cup \mathbb{Q}_4 \cup \mathbb{Q}_5 \cup \cdots\cdots$$

where $$\mathbb{Q}_n = \left\{ \frac{p}{q} \;\middle|\; p, q \geq 1,\ p+q = n \right\}$$

$$= \left\{ \frac{1}{n-1},\ \frac{2}{n-2},\ \cdots,\ \frac{n-1}{1} \right\}.$$

Ordering of $\mathbb{Q}$ can be obtained by adding 0 in the beginning and pairing every $a \in \mathbb{Q}^+$ with its negative $-a$.

Ordering of $\mathbb{Q}^+$ as above can be visualized as the zig-zag traversing of the infinite matrix as below.



Now we prove Cantor's Theorem

Theorem The set of real numbers in the interval $[0,1)$ is uncountable (and hence so is the set of all real numbers).

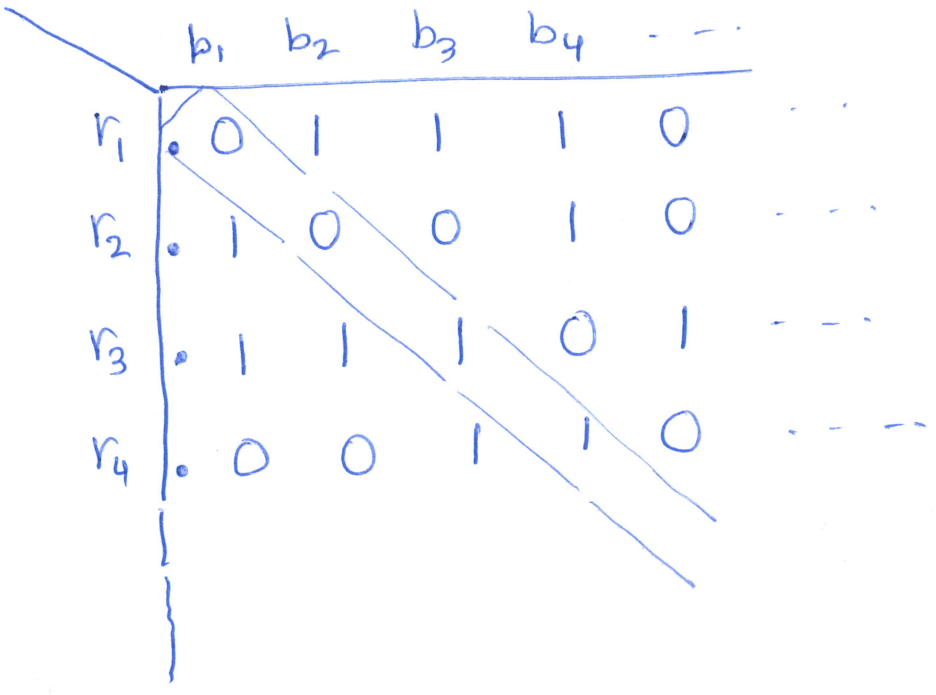
Proof Suppose on the contrary that there exists an ordering of all reals in $[0,1)$.

$$[0,1) = \{ r_1, r_2, r_3, r_4, \cdots \}.$$

We note that every real in $[0,1)$ has a binary representation of the form

$$0 \cdot b_1 b_2 b_3 b_4 \cdots \qquad b_i \in \{0,1\} \ \forall \ i \geq 1.$$

Construct an infinite matrix as below.



The rows of this matrix are indexed by the (supposedly ordered) reals $r_1, r_2, \ldots$ and each row is the (infinite) binary representation of the respective r_i, $i \geq 1$.

Now consider the real number r whose binary representation is the "diagonal of this matrix, but the entries flipped". I.e.

$$r = 0.c_1 c_2 c_3 c_4 \ldots$$

where for all $j \geq 1$,

$$c_j = \begin{cases} 1 & \text{if } j^{th} \text{ bit in } r_j \text{ is } 0 \\ 0 & \text{" } \quad 1. \end{cases}$$

Since r differs from r_j on j^{th} bit,

$$r \neq r_j.$$

This holds for every index $j \geq 1$ and thus r is distinct from every real in the ordering $\{r_1, r_2, r_3, r_4, \ldots\}$.

This contradicts the assumption that this

ordering lists every real in $[0,1)$!.

The contradiction proves that the set of reals in $[0,1)$ is actually uncountable. ▨

Note

① The proof technique is referred to as the "diagonalization method".

② The binary representations

$0.1000000\ldots$

$0.0111111\ldots$

both represent $\frac{1}{2}$. The representations are unique up to this ambiguity. For the sake of the proof, one can a priori agree to choose (say) the first representation in case of ambiguity.