XIX

NINETEENTH LECTURE
AUTOMATA & FORMAL LANGUAGES

28 November 2024

We encoded machines as binary strings:

$$M \longmapsto \mathrm{code}(M) \in \mathbb{B}$$

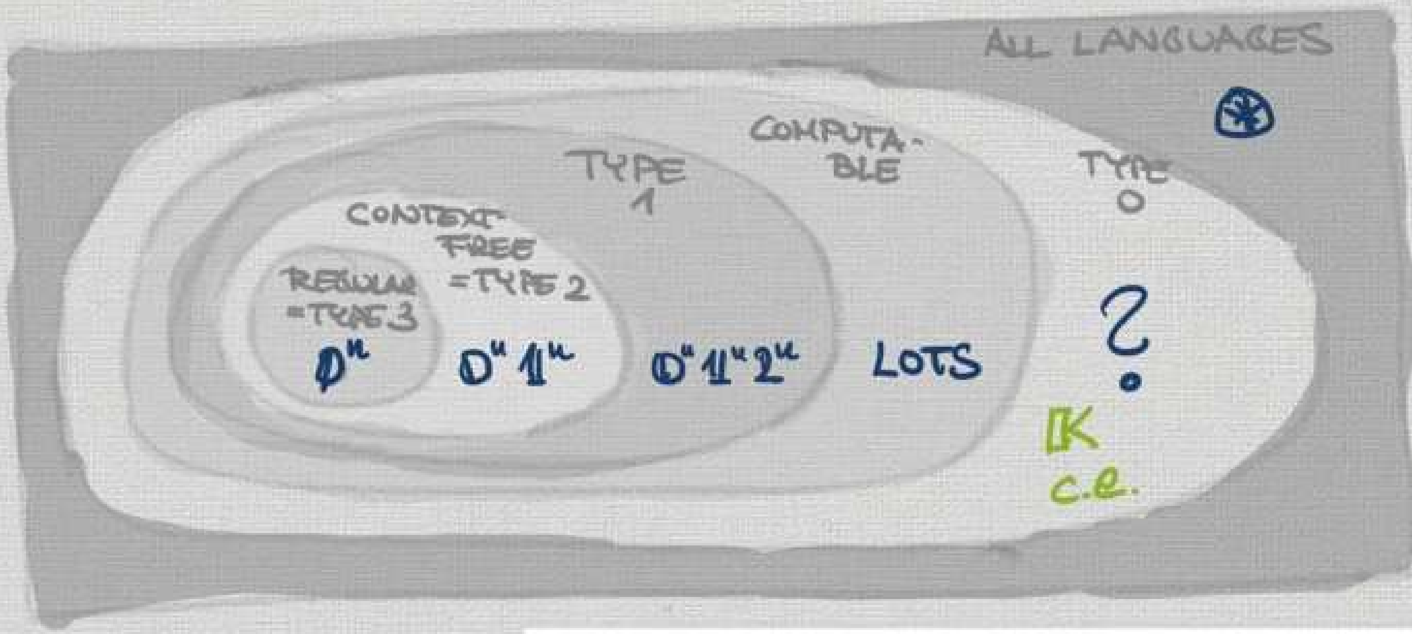
$$f_{w,k} : \mathbb{B}^k \dashrightarrow \mathbb{B}$$

is the k-ary function performed by M s.t. $\mathrm{code}(M) = w$

Uses the universal RM

$$W_w := \mathrm{dom}(f_{w,1})$$

Note that $\{W_w ; w \in \mathbb{B}\} = \{A \subseteq \mathbb{B} ; A \text{ is c.e.}\}$



REMINDER

Similarly, referring to numerical information can be used to refer to computation steps asking a given machine to run for a fixed number of steps; we call this *truncation* and it will play a very important role in §4.8. If M is a register machine and $k, n \in \mathbb{N}$, we can define sets $T_{M,k,n} \subseteq \mathbb{B}^k$, $T_{M,k} \subseteq \mathbb{B}^{k+1}$, and $\widehat{T}_{M,k} \subseteq \mathbb{B}^{k+2}$ as follows:

$$T_{M,k,n} := \{\vec{w} ; M \text{ has halted with input } \vec{w} \text{ after at most } n \text{ steps}\},$$
$$T_{M,k} := \{(\vec{w}, u) ; M \text{ has halted with input } \vec{w} \text{ after at most } \#u \text{ steps}\}, \text{ and}$$
$$\widehat{T}_{M,k} := \{(\vec{w}, u, v) ; (\vec{w}, u) \in T \text{ and } v \text{ is the content of register 0 at time of halting}\}.$$

Proposition 4.17. The sets $T_{M,k,n}$, $T_{M,k}$, and $\widehat{T}_{M,k}$ are computable.

§4.8 C.e. sets

$$\mathbb{K} := \{ w ; f_{w,1}(w)\downarrow \}$$ (TURING'S) HALTING PROBLEM

$$\mathbb{K}_0 := \{ (w,v) ; f_{w,1}(v)\downarrow \}$$

Thm 4.29 Both $\mathbb{K}$ and $\mathbb{K}_0$ are c.e.

Proof. $\mathbb{K}_0 = \mathrm{dom}(f_{U,2})$. The operation $w \longmapsto (w,w)$ can be performed by RM. Thus $f(w) := f_{U,2}(w,w)$ is computable and $\mathbb{K} = \mathrm{dom}(f)$. q.e.d

Thm 4.30 Neither $\mathbb{K}$ nor $\mathbb{K}_0$ is computable.

(Turing) **Proof.** Same for $\mathbb{K}, \mathbb{K}_0$. Define

$$f(w) := \begin{cases} \uparrow & \text{if } w \in \mathbb{K} \\ 0 & \text{if } w \notin \mathbb{K} \end{cases}$$

If $\mathbb{K}$ computable, then f is partial computable

CANTOR'S DIAGONAL ARGUMENT

Let d be s.t. $f_{d,1} = f$. $f_{d,1}(d)\downarrow \iff d \in \mathbb{K} \iff f(d)\uparrow \iff f_{d,1}(d)\uparrow$. Contradiction!

q.e.d

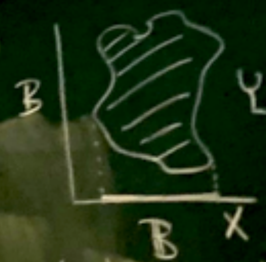
Hofstadter : "Limitative Theorems"

Def. Say $X \subseteq \mathbb{B}^n$ is Σ_1 if there is $Y \subseteq \mathbb{B}^{n+1}$ computable s.t.

$$\vec{w} \in X \iff \exists v\, (\vec{w}, v) \in Y$$

We call X Π_1 if it's the complement of a Σ_1 set.

Δ_1 if it's both Σ_1 and Π_1



Prop. 4.32 Every computable set is Δ_1.

Pf. Since computable set closed under complement, it's enough to show that every comp. set is Σ_1.

Fix X, define $Y := \{(\vec{w}, v), \vec{w} \in X, v \in \mathbb{B}\}$. Clearly computable

and $\vec{w} \in X \iff \exists v\, (\vec{w}, v) \in Y$. q.e.d

Thm 4.33 TFAE (i) X is c.e.
$X \subseteq \mathbb{B}^k$ (ii) X is Σ_1.

Proof (i) $\Rightarrow$ (ii). Let $X = \mathrm{dom}(f)$. Take M that computes f.

Consider $T_{M,k} = \{(\vec{w}, v) \,;\, f_{M,k}(\vec{w}) \text{ has halted after } \#v \text{ steps}\}$

$T_{M,k} \subseteq \mathbb{B}^{k+1}$ computable by P 4.17

Clearly $\vec{w} \in \mathrm{dom}(f) \iff \underbrace{\exists v \; (\vec{w}, v) \in T_{M,k}}_{\Sigma_1}$.

(ii) $\Rightarrow$ (i). Suppose X Σ_1, i.e., $Y \subseteq \mathbb{B}^{k+1}$ s.t.

$\vec{w} \in X \iff \exists v \; (\vec{w}, v) \in Y$. Y computable

Let $h : \mathbb{B}^k \dashrightarrow \mathbb{B}$ be the minimisation of χ_Y, i.e.,

$$h(\vec{w}) := \begin{cases} \text{least } v \text{ s.t. } (\vec{w}, v) \in Y & \text{if exists} \\ \uparrow & \text{o/w} \end{cases}$$

ES#3: h is computable

Clearly, $\mathrm{dom}(h) = \{\vec{w} \,;\, \exists v \; (\vec{w}, v) \in Y\} = X$.

q.e.d.

THE ZIGZAG METHOD

- Parallelisation
- Folding two quantifiers into one

Given $f: \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}$ and $\vec{w} \in \mathbb{B}^n$, is there $v \in \mathbb{B}^*$ s.t. $f(\vec{w}, v)\downarrow$?

	ε	$\mathbf{0}$	$\mathbf{1}$	$\mathbf{00}$	$\mathbf{10}$	$\mathbf{01}$	$\mathbf{11}$	$\mathbf{000}$ $\cdots$
ε	$(\vec{w}, \varepsilon, \varepsilon)$	$(\vec{w}, \varepsilon, \mathbf{0})$	$(\vec{w}, \varepsilon, \mathbf{1})$	$(\vec{w}, \varepsilon, \mathbf{00})$	$(\vec{w}, \varepsilon, \mathbf{10})$ $\cdots$			
$\mathbf{0}$	$(\vec{w}, \mathbf{0}, \varepsilon)$	$(\vec{w}, \mathbf{0}, \mathbf{0})$	$(\vec{w}, \mathbf{0}, \mathbf{1})$	$(\vec{w}, \mathbf{0}, \mathbf{00})$	$(\vec{w}, \mathbf{0}, \mathbf{10})$ $\cdots$			
$\mathbf{1}$	$(\vec{w}, \mathbf{1}, \varepsilon)$	$(\vec{w}, \mathbf{1}, \mathbf{0})$	$(\vec{w}, \mathbf{1}, \mathbf{1})$	$(\vec{w}, \mathbf{1}, \mathbf{00})$	$(\vec{w}, \mathbf{1}, \mathbf{10})$ $\cdots$			
$\mathbf{00}$	$(\vec{w}, \mathbf{00}, \varepsilon)$	$(\vec{w}, \mathbf{00}, \mathbf{0})$	$(\vec{w}, \mathbf{00}, \mathbf{1})$	$(\vec{w}, \mathbf{00}, \mathbf{00})$	$\ddots$			
$\mathbf{10}$	$(\vec{w}, \mathbf{10}, \varepsilon)$	$(\vec{w}, \mathbf{10}, \mathbf{0})$	$(\vec{w}, \mathbf{10}, \mathbf{1})$	$\ddots$				
$\vdots$	$\vdots$	$\vdots$	$\vdots$					

$$X := \{\vec{w} \,;\, \exists v \exists u \ (\vec{w}, v, u) \in Y\} \subseteq \mathbb{B}^k$$

<u>Claim</u> If Y is computable, X is c.e.

$$Z := \{(\vec{w}, v) \,;\, (\vec{w}, v_{(0)}, v_{(1)}) \in Y\}$$

Then $$\vec{w} \in X \iff \underbrace{\exists v \ (\vec{w}, v) \in Z}_{\Sigma_1 \Rightarrow \text{c.e.}}$$

Back to the example:

$$X = \{\vec{w}\,;\ \exists v\ f(\vec{w},v)\downarrow\} \text{ is c.e.}$$

Consider $T_{M,k+1} \subseteq \mathbb{B}^{k+2}$ and

$$\vec{w} \in X \iff \exists v\, \exists u\ (\vec{w},v,u) \in T_{M,k+1}.$$

By Zigzag Method, two quantifiers can be rolled into one, so X is Σ_1, thus X is c.e.

Similarly, referring to numerical information can be used to refer to computation steps asking a given machine to run for a fixed number of steps; we call this *truncation* and it will play a very important role in §4.8. If M is a register machine and $k, n \in \mathbb{N}$, we can define sets $T_{M,k,n} \subseteq \mathbb{B}^k$, $T_{M,k} \subseteq \mathbb{B}^{k+1}$, and $\widehat{T}_{M,k} \subseteq \mathbb{B}^{k+2}$ as follows:

$$\begin{aligned}
T_{M,k,n} &:= \{\vec{w}\,;\ M \text{ has halted with input } \vec{w} \text{ after at most } n \text{ steps}\},\\
T_{M,k} &:= \{(\vec{w},u)\,;\ M \text{ has halted with input } \vec{w} \text{ after at most } \#u \text{ steps}\}, \text{ and}\\
\widehat{T}_{M,k} &:= \{(\vec{w},u,v)\,;\ (\vec{w},u) \in T \text{ and } v \text{ is the content of register 0}\\
&\qquad \text{at time of halting}\}.
\end{aligned}$$

Proposition 4.17. The sets $T_{M,k,n}$, $T_{M,k}$, and $\widehat{T}_{M,k}$ are computable.

NOTE THAT THERE WAS A VARIABLE CONFUSION IN THE PROOF OF P 4.35 ON THE BLACKBOARD. THIS IS THE CORRECTION:

Proposition 4.35 Let $X \subseteq \mathbb{B}$, $X \neq \emptyset$.

Then TFAE

(i) X is c.e.

(ii) $\exists g$ computable $X = \mathrm{ran}(g)$.

Proof. "$\Longrightarrow$" If $X = \mathrm{dom}(f)$, let

$$g(w) := \begin{cases} w & \text{if } f(w)\downarrow \\ \uparrow & \text{o/w} \end{cases}$$

THINK WHY THIS IS COMPUTABLE
NOTE THAT THE CASE DISTINCTION LEMMA DOES NOT APPLY.

Then $\mathrm{ran}(g) = \mathrm{dom}(f) = X$.

"$\Longleftarrow$" Let $X = \mathrm{ran}(g)$. Let M compute g.

Consider $\hat{T}_{M,1}$ and observe

$$w \in X \Longleftrightarrow \underbrace{\exists v \exists u\, (v, u, w) \in \hat{T}_{M,1}}.$$

This is Σ_1 by zigzag method, so c.e. by T 4.33. q.e.d.