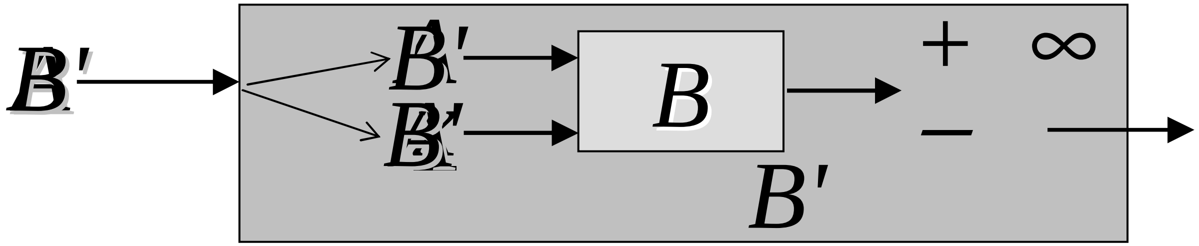
I. Recap of Discrete Computability

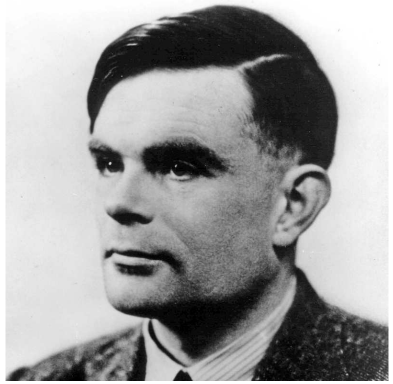
- un-/computability, Halting Problem
- Semi-/decidability, Reduction
- Model of computation: WHILE prog.
- SMN propery, Currying
- "Oracle" computation,
  Limit Lemma, Arithmetic Hierarchy

Alan M. Turing 1936

CS700 M. Ziegler

- first scientific calculations on digital computers
- *What are its fundamental limitations?*





- Undecidable Halting Problem *H*: **No** algorithm *B* can always correctly ans ~~simulator/interpreter *B*~~ ?

Given ⟨A,x⟩, does algorithm A terminate on input x?

Proof by contradiction: Consider algorithm *B*' that, on input *A*, executes *B* on ⟨*A*,*A*⟩ and, upon a positive answer, loops infinitely. How does *B*' behave on *B*' ?

Un-/Semi-/Decidability I

Definition: a) An 'algorithm' $\mathcal{A}$ computes a partial function $f :\subseteq \mathbb{N} \to \mathbb{N} = \{0,1,2,\ldots\}$ if it

- on inputs $\underline{x} \in \mathrm{dom}(f)$ outputs $f(\underline{x})$ and terminates,
- on inputs $\underline{x} \notin \mathrm{dom}(f)$ does not terminate.

Injective pairing function ("*Hilbert Hotel*")

$$\langle x,y\rangle := x + (x+y)\cdot(x+y+1)/2$$

b) $\mathcal{A}$ decides set $L \subseteq \mathbb{N}$ if it computes its total char. function: $\mathrm{cf}_L(\underline{x}):=1$ for $\underline{x} \in L$, $\mathrm{cf}_L(\underline{x}):=0$ for $\underline{x} \notin L$.

c) $\mathcal{A}$ semi-decides L if terminates precisely on $\underline{x} \in L$

d) $\mathcal{A}$ enumerates L if it computes some total injective $f:\mathbb{N}\to\mathbb{N}$ with $L=\mathrm{range}(f)$.

Un-/Semi-/Decidability II

Example: The Halting problem H, considered as subset of $\mathbb{N}$, is semi-decidable, not decidable.

Theorem: a) Every finite L is decidable.
b) L is decidable iff its complement $\overline{L}$ is.
c) L is decidable iff both $L, \overline{L}$ are semi-decidable.
d) L is enumerable iff infinite and semi-decidable.

b) $\mathcal{A}$ decides set $L\subseteq\mathbb{N}$ if it computes its total char. function: $\text{cf}_L(\underline{x}):=1$ for $\underline{x}\in L$, $\text{cf}_L(\underline{x}):=0$ for $\underline{x}\notin L$.

c) $\mathcal{A}$ semi-decides L if terminates precisely on $\underline{x}\in L$

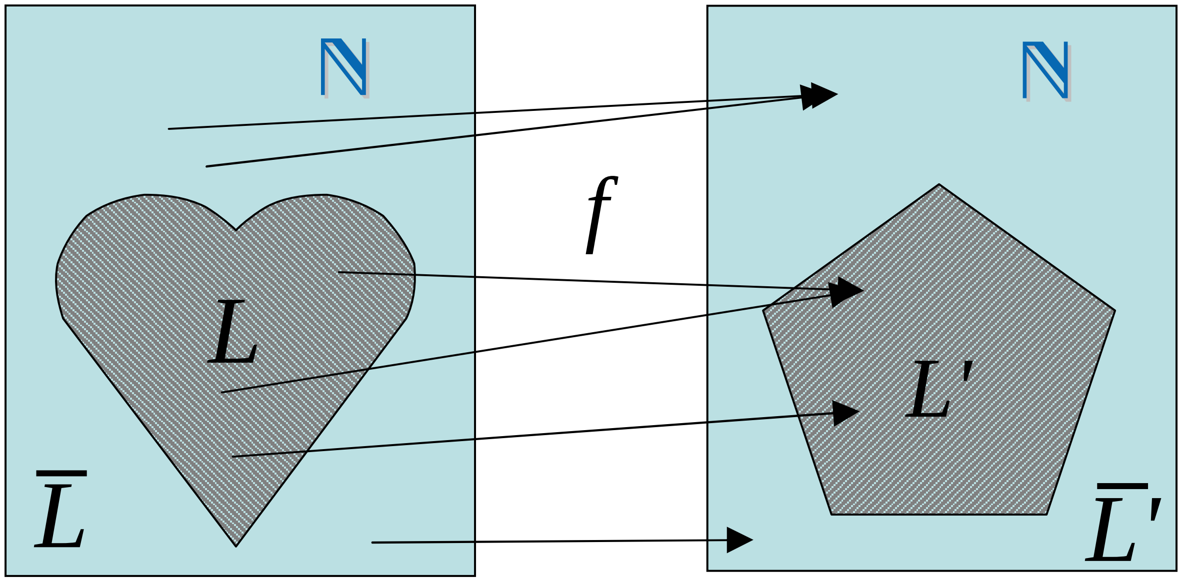
d) $\mathcal{A}$ enumerates L if it computes some total injective $f:\mathbb{N}\to\mathbb{N}$ with $L=\text{range}(f)$.

Comparing Decision Problems

Halting problem $H = \{ \langle \mathcal{A}, \underline{x} \rangle : \mathcal{A}(\underline{x})$ terminates $\}$

Nontriviality $N = \{ \langle \mathcal{A} \rangle : \exists y\ \mathcal{A}(\underline{y})$ terminates $\}$

Totality problem $T = \{ \langle \mathcal{A} \rangle : \forall \underline{z}\ \mathcal{A}(\underline{z})$ terminates$\}$



- $H \leqslant N$ — undecidable
- $H \leqslant T$ — undecidable
- $N \leqslant H \not\leqslant \overline{H}$
- $\overline{H} \leqslant T \Rightarrow T \not\leqslant H$

For $L, L' \subseteq \mathbb{N}$ write $L \leqslant L'$ if there is a computable $f: \mathbb{N} \to \mathbb{N}$ such that $\forall \underline{x}: \ \underline{x} \in L \ \Leftrightarrow \ f(\underline{x}) \in L'$.

a) $\overline{L}'$ semi-/decidable $\Rightarrow$ so $\overline{L}$. b) $L \leqslant L' \leqslant L'' \Rightarrow L \leqslant L''$

WHILE Programs

Syntax in Backus—Naur Form:

$$\mathcal{P} := (\ x_j := 0 \mid x_j := 1 \mid x_j := x_i + x_k \mid x_j := x_i - x_k$$
$$\mid x_j := x_i \div 2 \mid \mathcal{P} ; \mathcal{P} \mid \text{WHILE } x_j \text{ DO } P \text{ END}\ )$$

Semantics: loop executed as long as $x_j \neq 0$

Definition: Let $\langle\mathcal{P}\rangle \in \mathbb{N}$ denote the encoding of WHILE program $\mathcal{P}$ (e.g. as ascii sequence).

UTM-Theorem: There exists a WHILE program $\mathcal{U}$ that, given $\langle\mathcal{P}\rangle \in \mathbb{N}$ and $\langle x_1,\ldots,x_k\rangle \in \mathbb{N}$ and $N \in \mathbb{N}$, simulates $\mathcal{P}$ on input $(x_1,\ldots,x_k)$ for N steps.

SMN Theorem: Currying

Definition: Let $C = \langle \mathcal{P} \rangle \in \mathbb{N}$ denote the encoding of WHILE program $\mathcal{P}$, $\mathcal{P} = \rangle C \langle$ its inverse/decoding.

Type conversion **example**

$f(x,y) = \sin(x) \cdot e^y$

SMN-Theorem: There exists a WHILE program that, given $\langle \mathcal{P} \rangle \in \mathbb{N}$ and $x \in \mathbb{N}$, returns $\langle \mathcal{P}(x, \cdot\,) \rangle$, where $\mathcal{P}(x, \cdot\,)(y) :\equiv \mathcal{P}(x,y)$

UTM-Theorm: There exists a WHILE program that, given $\langle \mathcal{P} \rangle \in \mathbb{N}$, returns $\langle \mathcal{Q} \rangle \in \mathbb{N}$ with $\mathcal{Q}(x,y) = \rangle \mathcal{P}(x) \langle\, (y)$

and WHILE prg that, given $\langle \mathcal{P} \rangle, \langle \mathcal{Q} \rangle$, returns $\langle \mathcal{Q} \circ \mathcal{P} \rangle$

Oracle Computation

$\mathcal{P}^{\varphi} = (\, x_j := 0 \,,\, 1 \mid x_j := x_i \pm x_k \mid x_j := x_i \div 2 \mid$
$x_j := \varphi(x_i) \mid \mathcal{P} \,;\, \mathcal{P} \mid \text{WHILE } x_j \text{ DO } P \text{ END })$

Fix some *arbitrary* total $\varphi:\mathbb{N}\to\mathbb{N}$

Write H as short for cf_H

Lemma ("Shoenfield"): $g:\subseteq\mathbb{N}\to\mathbb{N}$ is HH-computable iff $g(x)=\lim_j f(\langle x,j\rangle)$ for some H-computable $f:\subseteq\mathbb{N}\to\mathbb{N}$.

Def: b) $\mathcal{P}^{\varphi}$ decides set $L\subseteq\mathbb{N}$ if it computes its total char. function: $\mathrm{cf}_L(\underline{x}):=1$ for $\underline{x}\in L$, $\mathrm{cf}_L(\underline{x}):=0$ for $\underline{x}\notin L$.

c) $\mathcal{P}^{\varphi}$ semi-decides L if terminates <u>precisely</u> on $\underline{x}\in L$

d) $H^{\varphi} = \{\, \langle \mathcal{P},\underline{x}\rangle : \mathcal{P}^{\varphi} \text{ terminat.on input } \underline{x} \,\}$ $\qquad H' := H^H$

Proof (Sketch) Shoenfield

$\mathcal{P}^{\varphi} = (x_j := 0 , 1 \mid x_j := x_i \pm x_k \mid x_j := x_i \div 2 \mid$
$x_j := \varphi(x_i) \mid \mathcal{P} ; \mathcal{P} \mid \text{WHILE } x_j \text{ DO } P \text{ END})$

Let $\mathcal{P}^H$ compute g. Fix x and initialize $A=\{\}=B$.

At each stage $j=1,2,\ldots$ maintain: $A\subseteq H$ and purportedly $B\cap H=\{\}$.

Lemma ("Shoenfield"): $g{:}\subseteq\mathbb{N}\to\mathbb{N}$ is H-computable iff $g(x)=\lim_j f(\langle x,j\rangle)$ for some computable $f{:}\subseteq\mathbb{N}\to\mathbb{N}$.

Simulate $\mathcal{P}$ on input x for j steps, answering all queries $y\in A$ positive and $y\in Q$ negative. Any query y neither in A nor in B tentatively add to B.

If simulation of $\mathcal{P}$ terminates, stop the current stage j and restart for $j+1$.

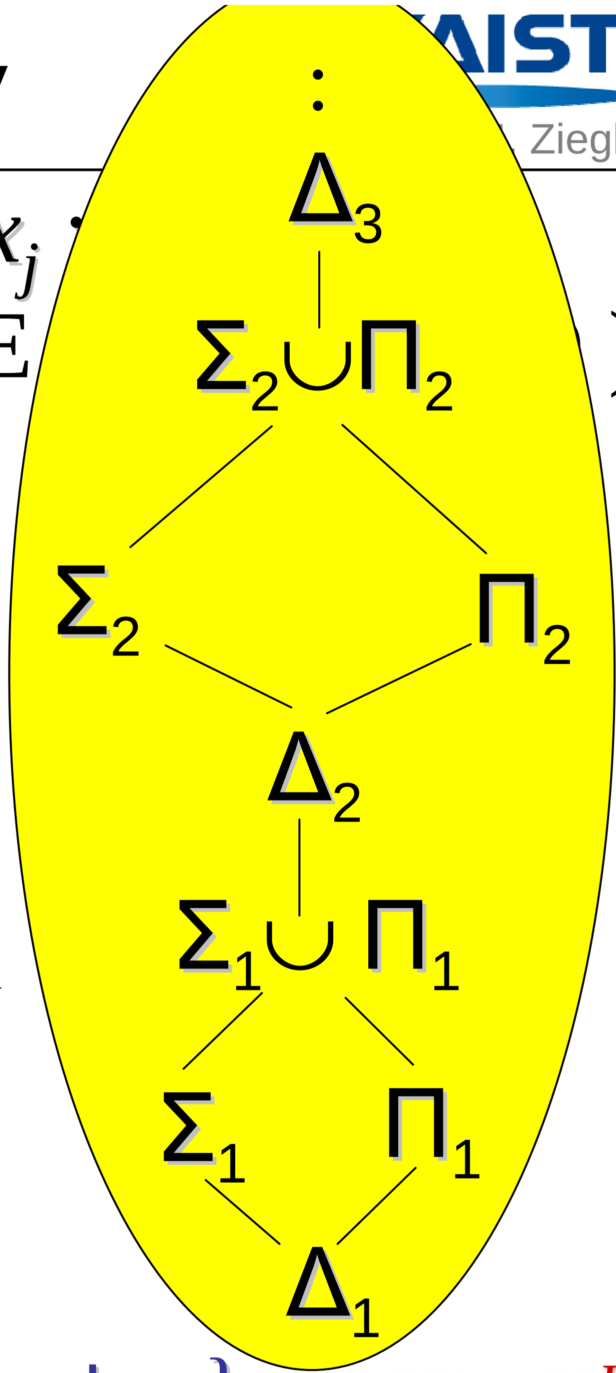
Simultaneously to simulating $\mathcal{P}$, semi-decide "$y\in H$?" for all $y\in B$..

When $y\in H$, move y from B to A, abort current stage j and restart for $j+1$.

Arithmetic Hierarchy

$\mathcal{P}^{\varphi} = (x_j := 0 , 1 \mid x_j := x_i \pm x_k \mid x_j$

$x_j := \varphi(x_i) \mid \mathcal{P} ; \mathcal{P} \mid$ WHILE)



Lemma: a) $\Delta_k = \text{co-}\Delta_k$

b) $\Delta_k = \Sigma_k \cap \Pi_k$

c) $\Sigma_k \cup \Pi_k \subsetneq \Delta_{k+1}$

Def: $\Delta_1 = \Sigma_0 = \Pi_0 =$ decidable

$\Delta_{k+1} = \text{decidable}^{\Sigma_k} = \text{decidable}^{\Pi_k}$

$\Sigma_{k+1} = \text{semi-decidable}^{\Sigma_k}$

$\Pi_{k+1} = \text{co-semi-decidable}^{\Sigma_k}$

d) $H^{\varphi} = \{ \langle \mathcal{P}, \underline{x} \rangle : \mathcal{P}^{\varphi}$ terminat.on input $\underline{x}$ $\}$ $H' := H^{H}$

I. Recap of Discrete Computability

- un-/computability, Halting Problem
- Semi-/decidability, Reduction
- Model of computation: WHILE prog.
- SMN propery, Currying
- "Oracle" computation,
  Limit Lemma, Arithmetic Hierarchy