CS125	Lecture 17	Fall 2016

17.1 The Halting Problem

Consider the HALTING PROBLEM (HALT_{TM}): Given a TM M and w, does M halt on input w?

Theorem 17.1 HALT_{TM} *is undecidable.*

Proof:

Suppose $\text{HALT}_{\text{TM}} = \{\langle M, w\rangle : M \text{ halts on } w\}$ were decided by some TM H.

Then we could use H to decide A_{TM} as follows.

On input $\langle M, w\rangle$,

- Modify M so that whenever it is about to halt in a rejecting configuration, it instead goes into an infinite loop. Call the resulting TM M'.
- Run $H(\langle M', w\rangle)$ and do the same.

Note that M' halts on w iff M accepts w, so this is indeed a decider for A_{TM}. $\Rightarrow\Leftarrow$.

$\blacksquare$

Proposition 17.2 *The* HALTING PROBLEM *is undecidable even for a fixed TM. That is, there is a TM M_0 such that ${\text{HALT}_{\text{TM}}}^{M_0} = \{w : M_0 \text{ halts on } w\}$ is undecidable.*

Proof: We define M_0 as follows: on input x, it attempts to interpret x as an encoding $\langle M, w\rangle$ of a Turing Machine M and input w to M. If x is not of the correct format, then M_0 simply rejects. Otherwise, M_0 simulates M on w and outputs whatever M outputs. Then ${\text{HALT}_{\text{TM}}}^{M_0}$ is undecidable since, if we could decide it via some Turing Machine P, then we could decide HALT_{TM}. In particular, if $\langle M, w\rangle$ were an input to HALT_{TM}, we would simply simulate P on $\langle M, w\rangle$. $\blacksquare$

Proposition 17.3 *The* HALTING PROBLEM *is undecidable even if we fix $w = \varepsilon$. That is, the language $\text{HALT}^{\varepsilon}_{\text{TM}} = \{\langle M\rangle : M \text{ halts on } \varepsilon\}$ is undecidable.*

Proof:

Suppose M_1 decided $\{\ \langle M\rangle\text{: } M \text{ halts on } \varepsilon\ \}$.
Then M_1 could be used to decide HALT_{TM}:

Given $\langle M, w\rangle$,

Construct $\langle M_w \rangle$, where M_w is a TM that writes
w on the empty tape and then runs M.
Then run M_1 on input $\langle M_w \rangle$

M_1 halts on $\langle M_w \rangle \Leftrightarrow M_w$ halts on $\varepsilon \Leftrightarrow M$ halts on w

But HALT_{TM} is undecidable. $\Rightarrow\Leftarrow$

■

Q: What if we fix both M and w?

17.2 Mapping Reductions

Definition 17.4 *A (mapping) reduction of $L_1 \subseteq \Sigma_1^*$ to $L_2 \subseteq \Sigma_2^*$ is a computable function $f : \Sigma_1^* \to \Sigma_2^*$ such that, for every $w \in \Sigma^*$,*

$w \in L_1$ *iff* $f(w) \in L_2$

We write $L_1 \leq_m L_2$.

(Note that here we do not require that f is computable in polynomial time.)

Lemma 17.5 *If $L_1 \leq_m L_2$, then*

- *if L_2 is decidable (resp., r.e.), then so is L_1;*
- *if L_1 is undecidable (resp., non-r.e.), then so is L_2.*

Examples:

- $A_{\text{TM}} \leq_m A_{\text{WR}}$ and $A_{\text{WR}} \leq_m A_{\text{TM}}$.

- For every Turing-recognizable (=r.e.) L, $L \leq_m A_{\text{TM}}$ (so A_{TM} is “r.e.-complete”).

- $A_{\text{TM}} \leq_m \text{HALT}_{\text{TM}}$.

- $\mathrm{HALT}_{\mathrm{TM}} \leq_m \mathrm{HALT}^{\varepsilon}_{\mathrm{TM}}$.

17.3 Rice's Theorem

Informally, Rice's Theorem says every (nontrivial) of the r.e. languages is undecidable.

Theorem 17.6 (Rice's Theorem) *Let $\mathcal{P}$ be any subset of the class of r.e. languages such that $\mathcal{P}$ and its complement are both nonempty. Then the language $L_{\mathcal{P}} = \{\langle M\rangle : L(M) \in \mathcal{P}\}$ is undecidable.*

Thus, given a TM (or Word-RAM program) M, it is undecidable to tell if

- $L(M) = \emptyset$,
- $L(M)$ is regular,
- $|L(M)| = \infty$, etc.

Proof:

- We will reduce $\mathrm{HALT}^{\varepsilon}_{\mathrm{TM}}$ to $L_{\mathcal{P}}$.
- Suppose without loss of generality that $\emptyset \notin \mathcal{P}$.
- Pick any $L_0 \in \mathcal{P}$ and say $L_0 = L(M_0)$.
- Define $f(\langle M\rangle) = \langle M'\rangle$, where

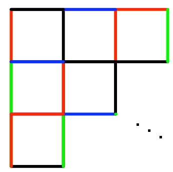
  M' is TM that on input w,
  - · first simulates M on input ε
  - · then simulates M_0 on input w
- **Claim:** f is a mapping reduction from $\mathrm{HALT}^{\varepsilon}_{\mathrm{TM}}$ to $L_{\mathcal{P}}$.
- Since $\mathrm{HALT}^{\varepsilon}_{\mathrm{TM}}$ is undecidable, so is $L_{\mathcal{P}}$.

■

17.4 Tiling

TILING: Given a finite set of patterns for square tiles:

Is it possible to tile the whole plane with tiles of these patterns in such a way that the abutting edges match?



Theorem 17.7 TILING *is undecidable.*

The proof of this theorem is quite involved for the general, unconstrained version of tiling. Instead, we'll prove undecidability for a variant of tiling, where we fix the tile at the origin and ask whether the first quadrant can be tiled.

Proof: We'll reduce from $\overline{\text{HALT}^{\varepsilon}_{\text{TM}}}$.

- $\langle M \rangle \overset{f}{\mapsto}$ sets of tiles so that:

  M does not halt on $\varepsilon \Leftrightarrow f(\langle M \rangle)$ tiles the first quadrant.

- View computation of M as "tableau", filling first quadrant with elements of $C = Q \cup \Gamma \cup \{\#\}$, each row being a configuration of M, except we fill the bottom-most row and left-most column of the tableau with #.

- Computation valid iff every 2×3 window consistent with transition function of M (and bottom row is correct initial configuration).

- Each tile represents a 2×3 window of tableau that is consistent with the transition function of M. So the set of tiles is a subset of C^6.

- Edge colors are $C^3 \cup C^4$, where C^3 is the set of colors used for the top and bottom edges of tiles (representing the two rows in a tile) and C^4 is used for the left and right edges of tiles (representing two overlapping 2x2 squares of the tile).

- Tile for origin is

#	q_0	$\sqcup$
#	#	#

  .

■

17.5 Diophantine Equations

Diophantine Equations are equations like

$$x^3y^3 + 13xyz = 4u^2 - 22$$

The coefficients and the exponents have to be integers. (No variables in the exponents!)

The question is whether the equation can be satisfied (made true) by substituting integers for the variables—this is known as Hilbert's 10th problem.

> "Given a diophantine equation with any number of unknown quantitites and with rational integral numerical coefficients: To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable in rational integers."

Diophantus of Alexandria (200–284 AD)

- "God gave him his boyhood one-sixth of his life, One twelfth more as youth while whiskers grew rife; And then yet one-seventh ere marriage begun; In five years there came a bouncing new son. Alas, the dear child of master and sage, after attaining half the measure of his father's life, chill fate took him. After consoling his fate by the science of numbers for four years, he ended his life."
- Other problems concerning triangular arrays, etc., gave rise to quadratic equations.
- Fermat's statement of his "Last Theorem" was in the margin of his copy of Diophantus.

Theorem 17.8 (Matiyasevich, 1970) *Hilbert's 10th problem is undecidable.*

Theorem 17.9 *A set $S \subseteq \mathbb{N}$ is r.e. iff it is of the form $\{x : (\exists y_1, y_2, \ldots, y_n)P(x, y_1, y_2, \ldots, y_n) = 0\}$ where P is a diophantine equation with $n+1$ variables ranging over $\mathbb{N}$.*

In fact, the theorem is true even with $n = 9$!

Other Undecidable Problems

- $\mathrm{ALL}_{\mathrm{CFG}}$: Given a context-free grammar G, is $L(G) = \Sigma^*$?
- The WORD PROBLEM for Finite Groups: Given a set of group generators $x_1, x_2, \ldots, x_n$ and a set R of relations between them (e.g. $x_1x_2 = x_2x_1, x_3 = x_1x_2^2x_3, \ldots$).

17.6 Undecidability and Gödel's Incompleteness Theorem

Fix an axiom systems for mathematics, e.g.

- Peano arithmetic — attempt to capture properties of $\mathbb{N}$
  E.g. $[\phi(0) \wedge (\forall n\,(\phi(n) \Rightarrow \phi(n+1)))] \Rightarrow \forall n\; \phi(n)$.
  What axiom is this?
- Zermelo-Frankel-Choice set theory (ZFC) — enough for all of modern mathematics
  E.g. $\forall y \exists z [\forall x (x \in z) \leftrightarrow (\forall w (w \in x) \rightarrow (w \in y))]$
  What axiom is this?

Proofs of theorems from these axiom systems defined by (simple) rules of mathematical logic.

From now on, we fix any axiom system that is:

- An extension of Peano arithmetic
- Sound & consistent: cannot prove false statements
- r.e. (e.g. there is a simple rule for listing the axioms).

Entscheidungsproblem is German for "Decision Problem." **The** Decision Problem is the problem of determining whether a mathematical statement is provable.

Proposition 17.10 *The set of all provable theorems is Turing-recognizable.*

Proof:

A proof is just a finite string: we could thus enumerate all finite strings and verify that they constitute a correct sequence of following the axioms, which leads to a proof of the desired statement.

Q: Is it decidable?

Theorem 17.11 (Church, Turing) *The set of all provable theorems is undecidable.*

Proof sketch:

- Reduce from $\mathrm{HALT}_{\mathrm{TM}}^{\varepsilon}$.
- $\langle M \rangle \mapsto$ mathematical statement ϕ_M = "$(\exists n)M$ halts on ε after n steps".
- **Claim:** M halts on ε iff ϕ_M is provable.

Theorem 17.12 (Gödel's Incompleteness Theorem) *There is a statement ϕ such that neither ϕ nor $\neg\phi$ is provable.*

Proof sketch:

- Suppose for contradiction that for all statements ϕ, either ϕ or $\neg\phi$ is provable. By consistency, both cannot be provable.

  $\Rightarrow$ Set of provable theorems r.e. and co-r.e.

  $\Rightarrow$ Set of provable theorems decidable.

- Contradiction.

Opening up the diagonalization and the reductions get explicit ϕ that essentially says "I am not provable".

Gödel's Letter to von Neumann, 1956: Can we decide in time $O(n)$ or $O(n^2)$ whether a mathematical statement has a proof of length n? If so, "it would obviously mean that in spite of the undecidability of the Entscheidungsproblem, the mental work of a mathematician concerning Yes-or-No questions could be completely replaced by a machine. . . ."

- This is an NP-complete problem!