Harvard CS 121 and CSCI E-207

Lecture 18: Reductions

Salil Vadhan

November 6, 2012

- Reading: Sipser §5.1, §5.3

Formalizing the Notion of Reduction

- L_1 "reduces" to L_2 if we can use a "black box" for L_2 to build an algorithm for L_1.

- A function $f : \Sigma_1^* \to \Sigma_2^*$ is computable if there is a Turing machine that for every input $w \in \Sigma_1^*$, M halts with just $f(w)$ on its tape.

- 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 any $w \in \Sigma^*$,

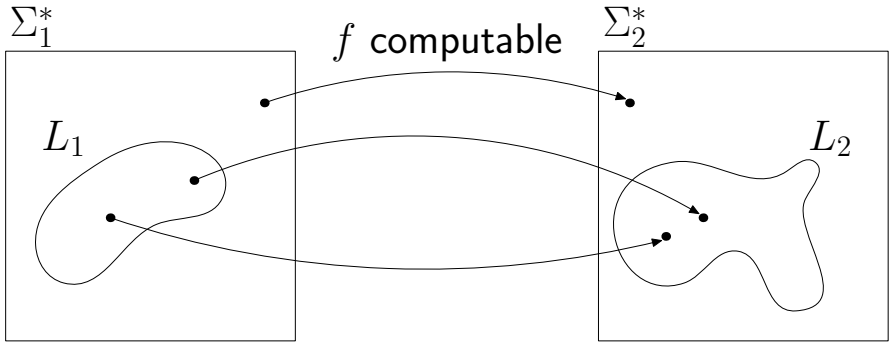
  $w \in L_1$ iff $f(w) \in L_2$

  We write $L_1 \leq_m L_2$.

Properties of Reducibility

Lemma: 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 of Reductions from Last Lecture

- For every Turing-recognizable L, $L \leq_m \mathrm{A}_{\mathrm{TM}}$.

- $\mathrm{A}_{\mathrm{TM}} \leq_m \mathrm{HALT}_{\mathrm{TM}}$.

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

Rice's Theorem

Informally: every (nontrivial) property of Turing-recognizable languages is undecidable.

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 M, it is undecidable to tell if

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

Proof of Rice's Theorem

- We will reduce $\text{HALT}^{\varepsilon}_{\text{TM}}$ 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 $\text{HALT}^{\varepsilon}_{\text{TM}}$ to $L_{\mathcal{P}}$.
- Since $\text{HALT}^{\varepsilon}_{\text{TM}}$ is undecidable, so is $L_{\mathcal{P}}$.

An Undecidable Problem about Context Free Grammars

Theorem: It is undecidable to determine, given CFGs G_1 and G_2, whether $L(G_1) \cap L(G_2) = \emptyset$.

Proof: Reduction from $\overline{A_{TM}}$, via "computation histories".

- Given $\langle M, w \rangle$, we can construct a CFG G_1 such that:

$$L(G_1) = \{C_1 \# D_1^R \# C_2 \# D_2^R \# \cdots \# C_n \# D_n^R : n \geq 1, \text{ the } C_i \text{ and } D_i \text{ are configurations of } M, \text{ and for each } i,\ C_i \Rightarrow_M D_i \}.$$

Intersection of CFLs, continued

- Similarly, we can construct a CFG G_2 such that

  $$L(G_2) = \{q_0 w \# C_1^R \# D_1 \# C_2^R \# D_2 \# \cdots \# C_n^R \# D_n \# u q_{\text{accept}} v : n \geq 0, \forall i\ C_i \text{ yields } D_i, u, v \in \Gamma^*\}.$$

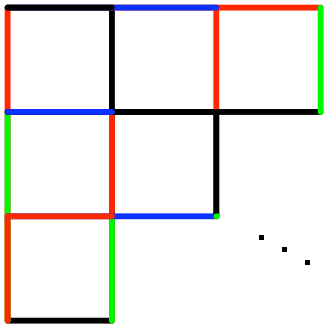
- Then $L(G_1) \cap L(G_2)$ is nonempty iff M accepts w.

Verifying computations is easier than carrying them out!

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: Tiling is undecidable.

Tiling, continued

Variant of tiling: fix the tile at the origin and ask whether the first quadrant can be tiled (easier to show undecidability).

Proof by reduction 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 $C = Q \cup \Gamma$, each row being a configuration of M.

- 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. Edge colors force consistency with neighbors on overlap.

Diophantine Equations

These 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.

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.

"Hilbert's 10th Problem"

10. Determination of the Solvability of a Diophantine Equation.

Given a diophantine equation with any number of unknown quantities 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.*

Thm (Matiyasevich, 1970): Hilbert's 10th problem is unsolvable.

Relation to Gödel's Incompleteness Theorem

Fix an axiom systems for mathematics, e.g.

- Peano arithmetic — attempt to capture properties of $\mathcal{N}$

  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

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

The Decision Problem (for Mathematics)

- **Entscheidungsproblem** is German for “Decision Problem”
- **The** Decision Problem is the problem of determining whether a mathematical statement is provable
- **Proposition:** Set of provable theorems is Turing-recognizable.

  Proof:

- **Q:** Is it decidable?

Undecidability of mathematics

Theorem [Church, Turing]: Set of provable statements of arithmetic (in any consistent extension of Peano arithmetic) is undecidable.

Proof sketch:

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

Incompleteness of Mathematics

Gödel's Incompleteness Theorem: There is a statement ϕ in arithmetic such that neither ϕ nor $\neg\phi$ is provable (in any consistent and r.e. extension of Peano arithmetic).

Proof sketch:

- Supoose 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.

Remark: Combined with previous proof, we see that it must be that a statement of the form $\neg\phi_M$ must be unprovable.

Coping with Undecidability

- Restrict to decidable special cases (e.g. quadratic diophantine equations).
- Use heuristics that are correct when they halt, but with no guarantee of halting on all inputs.
- Use programming languages whose syntactic structure makes it possible to detect or prevent certain kinds of bugs (e.g. type-safe languages).
- Formal verification: for small-scale programs, generate a formal proof that the program meets a formal specification (often with hints from programmer, such as loop invariants).