Lecture 18:
  • Mapping Reductions

Reading:
Sipser Ch 5.3

Mark Bun
November 15, 2022
Reductions

A **reduction** from problem $A$ to problem $B$ is an algorithm for problem $A$ which uses an algorithm for problem $B$ as a subroutine.

If such a reduction exists, we say “$A$ reduces to $B$”

**Positive uses:** If $A$ reduces to $B$ and $B$ is decidable, then $A$ is also decidable.

Ex. $E_{\text{DFA}}$ is decidable $\Rightarrow E_{Q_{\text{DFA}}}$ is decidable

**Negative uses:** If $A$ reduces to $B$ and $A$ is undecidable, then $B$ is also undecidable.

Ex. $E_{\text{TM}}$ is undecidable $\Rightarrow E_{Q_{\text{TM}}}$ is undecidable
What's wrong with the following "proof"?

Bogus "Theorem": \( A_{TM} \) is not Turing-recognizable

Bogus "Proof": Let \( R \) be an alleged recognizer for \( A_{TM} \). We construct a recognizer \( S \) for unrecognizable language \( \overline{A_{TM}} \):

On input \( \langle M, w \rangle \):
1. Run \( R \) on input \( \langle M, w \rangle \)
2. If \( R \) accepts, reject. Otherwise, accept.

If \( \langle M, w \rangle \in \overline{A_{TM}} \) when \( M \) loops on input \( w \) => \( R \) could loop on \( \langle M, w \rangle \) => \( S \) could loop on \( w \) => \( \langle M, w \rangle \notin \langle S \rangle \)

This sure looks like a reduction from \( \overline{A_{TM}} \) to \( A_{TM} \)
Mapping Reductions: Motivation

1. How do we formalize the notion of a reduction?
2. How do we use reductions to show that languages are unrecognizable?
3. How do we protect ourselves from accidentally “proving” bogus statements about recognizability?
Computable Functions

Definition:

A function $f: \Sigma^* \rightarrow \Sigma^*$ is **computable** if there is a TM $M$ which, given as input any $w \in \Sigma^*$, halts with only $f(w)$ on its tape. ("Outputs $f(w)$")
Computable Functions

Definition:
A function \( f : \Sigma^* \rightarrow \Sigma^* \) is computable if there is a TM \( M \) which, given as input any \( w \in \Sigma^* \), halts with only \( f(w) \) on its tape. ("Outputs \( f(w) \)")

Example 1: \( f(w) = \text{sort}(w) \) HW5 Problem 3

Example 2: \( f((x, y)) = x + y \)

Input: \( x_1, x_2, \ldots, x_n \# y_1, y_2, \ldots, y_m \)

Output: \( z_1, z_2, z_3, \ldots, z_m \)

where \( z = x + y \)
Computable Functions

Definition:

A function \( f : \Sigma^* \rightarrow \Sigma^* \) is **computable** if there is a TM \( M \) which, given as input any \( w \in \Sigma^* \), halts with only \( f(w) \) on its tape. ("Outputs \( f(w) \)"")

Example 3: \( f(\langle M, w \rangle) = \langle M' \rangle \) where \( M \) is a TM, \( w \) is a string, and \( M' \) is a TM that ignores its input and simulates running \( M \) on \( w \)

1. **Construct** TM \( M' \):
   - On input \( x \): Ignore \( x \), run \( M \) on \( w \). If it accepts, accept; if rejects, reject

2. **Output** \( \langle M' \rangle \)

11/15/2022

CS332 - Theory of Computation
Mapping Reductions

Definition:
Let $A, B \subseteq \Sigma^*$ be languages. We say $A$ is mapping reducible to $B$, written $A \leq_m B$, if there is a computable function $f: \Sigma^* \rightarrow \Sigma^*$ such that for all strings $w \in \Sigma^*$, we have $w \in A \iff f(w) \in B$. 
Mapping Reductions

Definition:
Language $A$ is mapping reducible to language $B$, written
$A \leq_m B$
if there is a computable function $f : \Sigma^* \to \Sigma^*$ such that for all strings $w \in \Sigma^*$, we have $w \in A \iff f(w) \in B$

If $A \leq_m B$, which of the following is true?

a) $\overline{A} \leq_m B$
b) $A \leq_m \overline{B}$
c) $\overline{A} \leq_m \overline{B}$
d) $\overline{B} \leq_m \overline{A}$
Decidability

Theorem: If \( A \leq_m B \) and \( B \) is decidable, then \( A \) is also decidable.

Proof: Let \( M \) be a decider for \( B \) and let \( f: \Sigma^* \rightarrow \Sigma^* \) be a mapping reduction from \( A \) to \( B \). Construct a decider \( N \) for \( A \) as follows:

On input \( w \):

1. Compute \( f(w) \)
2. Run \( M \) on input \( f(w) \)
3. If \( M \) accepts, accept. If it rejects, reject.

\[
\begin{align*}
\text{Correctness:} \\
1) & \text{ If } w \in A \implies f(w) \in B \quad [\text{by defn of mapping reduction}] \\
\implies & M \text{ accepts } f(w) \quad [M \text{ decides } B] \\
\implies & N \text{ accepts } \checkmark \\
2) & \text{ If } w \notin A \implies f(w) \notin B \quad [\text{by defn of mapping reduction}] \\
\implies & M \text{ rejects } f(w) \quad [M \text{ decides } B] \\
\implies & N \text{ rejects } \checkmark
\end{align*}
\]
Undecidability

Theorem: If $A \leq_m B$ and $B$ is decidable, then $A$ is also decidable.

Corollary: If $A \leq_m B$ and $A$ is undecidable, then $B$ is also undecidable.

(Contrapositive of Thm)
Old Proof: Equality Testing for TMs

\[ EQ_{TM} = \{ \langle M_1, M_2 \rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2) \} \]

Theorem: \( EQ_{TM} \) is undecidable

Proof: Suppose for contradiction that there exists a decider \( R \) for \( EQ_{TM} \). We construct a decider for \( E_{TM} \) as follows:

On input \( \langle M \rangle \):

1. Construct TMs \( M_1, M_2 \) as follows:
   \[ M_1 = M \]
   \[ M_2 = \text{"On input } x,\]
   \[ 1. \text{ Ignore } x \text{ and reject"} \]

2. Run \( R \) on input \( \langle M_1, M_2 \rangle \)
3. If \( R \) accepts, accept. Otherwise, reject.

This is a reduction from \( E_{TM} \) to \( EQ_{TM} \)
New Proof: Equality Testing for TMs

\[ EQ_{TM} = \{ \langle M_1, M_2 \rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2) \} \]

Theorem: \( E_{TM} \leq_{m} EQ_{TM} \) hence \( EQ_{TM} \) is undecidable

Proof: The following TM \( N \) computes the reduction \( f: \)

\[ \begin{align*}
\text{If } \langle M \rangle \in E_{TM} & \Rightarrow L(M) = \emptyset \quad \Rightarrow L(M_1) = L(M_2) = \emptyset \quad \Rightarrow \langle M_1, M_2 \rangle \in EQ_{TM} \\
\text{If } \langle M \rangle \notin E_{TM} & \Rightarrow L(M) \neq \emptyset \quad \Rightarrow L(M_1) \neq L(M_2) = \emptyset \quad \Rightarrow \langle M_1, M_2 \rangle \notin EQ_{TM}
\end{align*} \]

On input \( \langle M \rangle \):

1. Construct TMs \( M_1, M_2 \) as follows:
   \[ L(M_1) = \emptyset \]
   \[ M_1 = M \quad L(M_1) = L(M) \quad M_2 = \text{"On input } x, 1. \text{ Ignore } x \text{ and reject"} \]

2. Output \( \langle M_1, M_2 \rangle \)

---

Claim: \( f \) is a mapping reduction from \( E_{TM} \) to \( EQ_{TM} \)

\[ \begin{align*}
1) \quad & \langle M \rangle \in E_{TM} \Rightarrow f(\langle M \rangle) \in EQ_{TM} \\
2) \quad & \langle M \rangle \notin E_{TM} \Rightarrow f(\langle M \rangle) \notin EQ_{TM}
\end{align*} \]
Mapping Reductions: Recognizability

Theorem: If $A \leq_m B$ and $B$ is recognizable, then $A$ is also recognizable.

Proof: Let $M$ be a recognizer for $B$ and let $f : \Sigma^* \rightarrow \Sigma^*$ be a mapping reduction from $A$ to $B$. Construct a recognizer $N$ for $A$ as follows:

1. **Correctness:**
   1) If $w \in A \Rightarrow f(w) \in B$
   
   $\Rightarrow M$ accepts $\Rightarrow N$ accepts $\checkmark$

2) If $w \notin A \Rightarrow f(w) \notin B$

   $\Rightarrow M$ does not accept $\checkmark$

   $\Rightarrow N$ does not accept $\checkmark$

On input $w$:

1. Compute $f(w)$
2. Run $M$ on input $f(w)$
3. If $M$ accepts, accept. Otherwise, reject.
Unrecognizability

Theorem: If \( A \leq_m B \) and \( B \) is recognizable, then \( A \) is also recognizable.

Corollary: If \( A \leq_m B \) and \( A \) is unrecognizable, then \( B \) is also unrecognizable.

Corollary: If \( \overline{A_{TM}} \leq_m B \), then \( B \) is unrecognizable.

Corollary: If \( A_{TM} \leq_m \overline{B} \), then \( B \) is unrecognizable.

Why? \( A_{TM} \leq_m \overline{B} \iff \overline{A_{TM}} \leq_m B \)
Recognizability and $A_{TM}$

Let $L$ be a language. Which of the following is true?

a) If $L \leq_m A_{TM}$, then $L$ is recognizable

b) If $A_{TM} \leq_m L$, then $L$ is recognizable

\[\text{c) If } L \text{ is recognizable, then } L \leq_m A_{TM}\]

\[\text{d) If } L \text{ is recognizable, then } A_{TM} \leq_m L\]

\[\text{Lemma: } A_{TM} = \{ \langle m, w \rangle \mid \text{TM } m \text{ accepts } w \text{ is recognizable}\]

\[\text{Theorem: } L \text{ is recognizable if and only if } L \leq_m A_{TM}\]
Recognizability and $A_{TM}$

**Theorem:** $L$ is recognizable if and only if $L \leq_m A_{TM}$

**Proof:**

$\implies$ Let $L$ be recognizable. Goal: Construct mapping reduction from $L$ to $A_{TM}$.

Let $M$ be a $TM$ recognizing $L$.

The following $TM$ computes a mapping reduction from $L$ to $A_{TM}$:

- **Input:** $w$
- **Output:** $\langle M, w \rangle$

**Correctness:**
1) If $w \in L \implies \langle M, w \rangle \in A_{TM}$
2) If $w \notin L \implies M$ does not accept $w$
Example: Another reduction to $EQ_{TM}$

$EQ_{TM} = \{ \langle M_1, M_2 \rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2) \}$

Theorem: $A_{TM} \leq_m EQ_{TM}$

Proof: The following TM $N$ computes the reduction $f$:

What should the inputs and outputs to $f$ be?

a) $f$ should take as input a pair $\langle M_1, M_2 \rangle$ and output a pair $\langle M, w \rangle$

b) $f$ should take as input a pair $\langle M, w \rangle$ and output a pair $\langle M_1, M_2 \rangle$

c) $f$ should take as input a pair $\langle M_1, M_2 \rangle$ and either accept or reject

d) $f$ should take as input a pair $\langle M, w \rangle$ and either accept or reject
Example: Another reduction to $EQ_{TM}$

$$EQ_{TM} = \{\langle M_1, M_2 \rangle | M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$$

**Theorem:** $A_{TM} \leq_{m} EQ_{TM}$

**Proof:** The following TM computes the reduction $f$:

- If $\langle M, w \rangle \in A_{TM} \Rightarrow f(\langle M, w \rangle) = \langle M_1, M_2 \rangle \in EQ_{TM}$
- If $\langle M, w \rangle \notin A_{TM} \Rightarrow f(\langle M, w \rangle) = \langle M_1, M_2 \rangle \notin EQ_{TM}$

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

1. **Construct TMs $M_1, M_2$ as follows:**
   - $M_1 = "\text{On input } x, \text{Accept}"$
   - $M_2 = "\text{On input } x, \text{Run } M \text{ on } w. \text{ If accepted, accept. If rejected, reject}"$

2. Output $\langle M_1, M_2 \rangle$
Consequences of $A_{TM} \leq_m EQ_{TM}$

1. Since $A_{TM}$ is undecidable, $EQ_{TM}$ is also undecidable

2. $A_{TM} \leq_m EQ_{TM}$ implies $\overline{A_{TM}} \leq_m EQ_{TM}$, Since $\overline{A_{TM}}$ is unrecognizable, $\overline{EQ_{TM}}$ is unrecognizable
Theorem: $EQ_{TM}$ itself is also unrecognizable

$EQ_{TM} = \{\langle M_1, M_2 \rangle | M_1, M_2$ are TMs and $L(M_1) = L(M_2)\}$

Proof: The following TM computes the reduction:

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

1. Construct TMs $M_1, M_2$ as follows:
   - $M_1 = \text{"On input } x,\text{ accept."}$
     1. Ignore $x$
     2. Run $M$ on input $w$
     3. If $M$ accepts, accept. Otherwise, reject.
   - $M_2 = \text{"On input } x,\text{ reject.}"$
     1. Ignore $x$ and reject.

2. Output $\langle M_1, M_2 \rangle$