The Halting Problem

Sipser 4.2 (pages 173-182)

Taking stock
Are there problems a computer can't solve?! 

• But they seem so powerful...

• What about software verification?
  – Given a program and a specification of what it should do, can we check if it is correct?

What about deciding TMs?

• \( A_{TM} = \{ <M,w> \mid M \text{ is a TM and } M \text{ accepts } w \} \)
• Theorem 4.11: \( A_{TM} \) is undecidable!
• Is it even recognizable?
• Let \( U = \) “On input \( <M,w> \), where \( M \) is a TM:
  1. Simulate \( M \) on input \( w \).
  2. If \( M \) ever enters its accept state, accept;
     if \( M \) ever enters its reject state, reject.”
Towards proving undecidability

• Cantor 1873: How can we tell whether one infinite set is “larger” than another?

• A function $f$ from $A$ to $B$ is
  – One-to-one if $f(x) \neq f(y)$ if $x \neq y$
    • (f never maps two elements to the same value)
  – Onto if every element of $B$ is hit

• A correspondence is a function that is both one-to-one and onto

Countable sets

• Sets $A$ and $B$ have the same size if:
  – $A$ and $B$ are finite with the same number of elements
  – $A$ and $B$ are infinite with a correspondence between them

• A set is countable if it is finite or has the same size as $\mathbb{N}$
  – (natural numbers 1, 2, 3,...)
For example...

• $E = \{ \text{even natural numbers} \}$ is countable
• Define $f : \mathbb{N} \rightarrow E$ as $f(n) = 2n$

<table>
<thead>
<tr>
<th>n</th>
<th>f(n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Diagonalization

• Theorem 4.17: $\mathbb{R}$ is uncountable.
• Proof: 
  By contradiction. Assume there is a correspondence. We find a real number $x \neq f(n)$ for any natural number $n$.

<table>
<thead>
<tr>
<th>n</th>
<th>f(n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.14159...</td>
</tr>
<tr>
<td>2</td>
<td>55.55555...</td>
</tr>
<tr>
<td>3</td>
<td>0.12345...</td>
</tr>
<tr>
<td>4</td>
<td>0.50000...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Put your thinking caps on!

• How do we show that:

- $\Sigma^*$ is countable
  • assume $\Sigma = \{0, 1\}$

- $B = \{\text{all infinite binary sequences}\}$ is uncountable

Uh-oh...

• Theorem 4.18: Some languages are not Turing-recognizable.

• Proof:
  Let $L$ be the set of all languages over $\Sigma$. Define a correspondence from $L$ to $B$.

  $\Sigma^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, ...\}$
  $A = \{0, 00, 01, 000, 001, ...\}$
  $X_A = 0 1 0 1 1 0 0 1 1 ...$
The Halting Problem

- $A_{TM} = \{<M,w> \mid M \text{ is a TM and } M \text{ accepts } w\}$
- Theorem 4.11: $A_{TM}$ is undecidable.
- Proof:
  By contradiction. Assume $H$ is a decider for $A_{TM}$. We construct a TM
  $D = "\text{On input } <M>, \text{ where } M \text{ is a TM:}\$
  1. Run $H$ in input $<M, <M>>$
  2. If $H$ accepts, reject; if $H$ rejects, accept."

Huh?

$D(<M>) = \begin{cases} 
\text{accept} & \text{if } M \text{ does not accept } <M> \\
\text{reject} & \text{if } M \text{ accepts } <M>
\end{cases}$

- What if we run $D$ on $<D>$?
- Then
  $D(<D>) = \begin{cases} 
\text{accept} & \text{if } D \text{ does not accept } <D> \\
\text{reject} & \text{if } D \text{ accepts } <D>
\end{cases}$

- Contradiction! Then $H$ cannot exist and $A_{TM}$ is undecidable.
Where is the diagonalization?

• Running a machine on its description

<table>
<thead>
<tr>
<th></th>
<th>$&lt;M_1&gt;$</th>
<th>$&lt;M_2&gt;$</th>
<th>$&lt;M_3&gt;$</th>
<th>$&lt;M_4&gt;$</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>accept</td>
<td>accept</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_2$</td>
<td>accept</td>
<td>accept</td>
<td>accept</td>
<td>accept</td>
<td></td>
</tr>
<tr>
<td>$M_3$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_4$</td>
<td>accept</td>
<td>accept</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Where is the diagonalization?

• Running $H$ on a machine and its description

<table>
<thead>
<tr>
<th></th>
<th>$&lt;M_1&gt;$</th>
<th>$&lt;M_2&gt;$</th>
<th>$&lt;M_3&gt;$</th>
<th>$&lt;M_4&gt;$</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td>accept</td>
<td>reject</td>
<td>accept</td>
<td>reject</td>
<td></td>
</tr>
<tr>
<td>$M_2$</td>
<td>accept</td>
<td>accept</td>
<td>accept</td>
<td>accept</td>
<td></td>
</tr>
<tr>
<td>$M_3$</td>
<td>reject</td>
<td>reject</td>
<td>reject</td>
<td>reject</td>
<td></td>
</tr>
<tr>
<td>$M_4$</td>
<td>accept</td>
<td>accept</td>
<td>reject</td>
<td>reject</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Where is the diagonalization?

- Adding $D$ to the picture

<table>
<thead>
<tr>
<th></th>
<th>$&lt;M_1&gt;$</th>
<th>$&lt;M_2&gt;$</th>
<th>$&lt;M_3&gt;$</th>
<th>$&lt;M_4&gt;$</th>
<th>...</th>
<th>$&lt;D&gt;$</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M_1$</td>
<td><strong>accept</strong></td>
<td>reject</td>
<td>accept</td>
<td>reject</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_2$</td>
<td><strong>accept</strong></td>
<td><strong>accept</strong></td>
<td>accept</td>
<td>accept</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_3$</td>
<td>reject</td>
<td>reject</td>
<td><strong>reject</strong></td>
<td>reject</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$M_4$</td>
<td>accept</td>
<td>accept</td>
<td>reject</td>
<td><strong>reject</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$D$</td>
<td>reject</td>
<td>reject</td>
<td>accept</td>
<td>accept</td>
<td>?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

So have we found a language that is not Turing-recognizable?

- Theorem 4.22: A language is decidable iff it and its complement are recognizable

- Then $\overline{A_{TM}}$ is not Turing-recognizable!
Updating the picture

All languages

Turing-recognizable

Turing-decidable

Context-free languages

Regular languages

\(0^n 1^n\)

\(a^n b^n c^n\)

\(0^* 1^*\)

\(\overline{T_{TM}}\)