1 Rice's Theorem

1.1 Properties

Checking Properties

Given M

$$\left.\begin{array}{l}\text{Does } \mathbf{L}(M) \text{ contain } \langle M\rangle? \\ \text{Is } \mathbf{L}(M) \text{ non-empty?} \\ \text{Is } \mathbf{L}(M) \text{ empty?}\end{array}\right\} \text{Undecidable}$$

$$\left.\begin{array}{l}\text{Is } \mathbf{L}(M) \text{ infinite?} \\ \text{Is } \mathbf{L}(M) \text{ finite?} \\ \text{Is } \mathbf{L}(M) \text{ co-finite (i.e., is } \overline{\mathbf{L}(M)} \text{ finite)?} \\ \text{Is } \mathbf{L}(M) = \Sigma^*?\end{array}\right\} \text{Undecidable}$$

None of these properties can be decided. This is the content of Rice's Theorem.

Properties

Definition 1. A property of languages is simply a set of languages. We say L *satisfies* the property $\mathbb{P}$ if $L \in \mathbb{P}$.

Definition 2. For any property $\mathbb{P}$, define language $L_{\mathbb{P}}$ to consist of Turing Machines which accept a language in $\mathbb{P}$:

$$L_{\mathbb{P}} = \{\langle M\rangle \mid \mathbf{L}(M) \in \mathbb{P}\}$$

Deciding $L_{\mathbb{P}}$: deciding if a language represented as a TM satisfies the property $\mathbb{P}$.

- *Example:* $\{\langle M\rangle \mid \mathbf{L}(M) \text{ is infinite}\}$; $E_{\text{TM}} = \{\langle M\rangle \mid \mathbf{L}(M) = \emptyset\}$
- *Non-example:* $\{\langle M\rangle \mid M \text{ has 15 states}\}$ $\longleftarrow$ This is a property of TMs, and not languages!

Trivial Properties

Definition 3. A property is trivial if either it is not satisfied by any r.e. language, or if it is satisfied by all r.e. languages. Otherwise it is non-trivial.

Example 4. Some trivial properties:

- $\mathbb{P}_{\text{ALL}}$ = set of all languages
- $\mathbb{P}_{\text{R.E.}}$ = set of all r.e. languages
- $\overline{\mathbb{P}}$ where $\mathbb{P}$ is trivial
- $\mathbb{P} = \{L \mid L \text{ is recognized by a TM with an even number of states}\} = \mathbb{P}_{\text{R.E.}}$

Observation. For any trivial property $\mathbb{P}$, $L_{\mathbb{P}}$ is decidable. (Why?) Then $L_{\mathbb{P}} = \Sigma^*$ or $L_{\mathbb{P}} = \emptyset$.

1.2 Main Theorem

Rice's Theorem

Proposition 5. *If $\mathbb{P}$ is a non-trivial property, then $L_{\mathbb{P}}$ is undecidable.*

- Thus $\{\langle M\rangle \mid \mathbf{L}(M) \in \mathbb{P}\}$ is not decidable (unless $\mathbb{P}$ is trivial)

We cannot algorithmically determine any interesting property of languages represented as Turing Machines!

Properties of TMs

Note. Properties of TMs, as opposed to those of languages they accept, may or may not be decidable.

Example 6.

$$\left.\begin{array}{l}\{\langle M\rangle \mid M \text{ has 193 states}\} \\ \{\langle M\rangle \mid M \text{ uses at most 32 tape cells on blank input}\}\end{array}\right\} \text{Decidable}$$

$$\left.\begin{array}{l}\{\langle M\rangle \mid M \text{ halts on blank input}\} \\ \{\langle M\rangle \mid \text{ on input 0011 } M \text{ at some point writes the} \\ \qquad \text{symbol \$ on its tape}\}\end{array}\right\} \text{Undecidable}$$

Proof of Rice's Theorem

Rice's Theorem
If $\mathbb{P}$ is a non-trivial property, then $L_{\mathbb{P}}$ is undecidable.

Proof. Suppose $\mathbb{P}$ non-trivial and $\emptyset \notin \mathbb{P}$. If $\emptyset \in \mathbb{P}$, then in the following we will be showing $L_{\overline{\mathbb{P}}}$ is undecidable. Then $L_{\mathbb{P}} = \overline{L_{\overline{\mathbb{P}}}}$ is also undecidable.

Recall $L_{\mathbb{P}} = \{\langle M\rangle \mid \mathbf{L}(M) \text{ satisfies } \mathbb{P}\}$. We'll reduce A_{TM} to $L_{\mathbb{P}}$. Then, since A_{TM} is undecidable, $L_{\mathbb{P}}$ is also undecidable. Broadly the idea behind the reduction is as follows. Since $\mathbb{P}$ is non-trivial, at least one r.e. language satisfies $\mathbb{P}$. i.e., $\mathbf{L}(M_0) \in \mathbb{P}$ for some TM M_0. We will show a reduction f that maps an instance $\langle M, w\rangle$ for A_{TM}, to N such that

- If M accepts w then N accepts the same language as M_0. Then $\mathbf{L}(M) = \mathbf{L}(M_0) \in \mathbb{P}$
- If M does not accept w then N accepts $\emptyset$. Then $L(N) = \emptyset \notin \mathbb{P}$

Thus, $\langle M, w\rangle \in A_{\text{TM}}$ iff $N \in L_{\mathbb{P}}$.

We now describe the reduction precisely. The reduction f maps $\langle M, w\rangle$ to $\langle N\rangle$, where N is a TM that behaves as follows:

```
On input x
    Ignore the input and run M on w
    If M does not accept (or doesn't halt)
        then do not accept x (or do not halt)
    If M does accept w
        then run M0 on x and accept x iff M0 does.
```

Notice that indeed if M accepts w then $\mathbf{L}(N) = \mathbf{L}(M_0)$. Otherwise $\mathbf{L}(N) = \emptyset$. $\square$

Rice's Theorem
Recap

Every non-trivial property of r.e. languages is undecidable

- Rice's theorem says nothing about properties of Turing machines
- Rice's theorem says nothing about whether a property of languages is recurisvely enumerable or not.

Big Picture . . . again