Theory of Machines and Languages

Fatemeh Deldar

1403-1404

A Universal Turing Machine

- **Every Turing machine can be represented by a string of 0's and 1's**
  - **Assume that:**

$$Q = \{q_1, q_2, \ldots, q_n\} \qquad \Gamma = \{a_1, a_2, \ldots, a_m\}$$

  - **We select an encoding in which**
    - q_1 **is represented by 1**
    - q_2 **is represented by 11, and so on**
    - a_1 **is encoded as 1**
    - a_2 **as 11, etc**
    - **The symbol 0 will be used as a separator between the 1's**

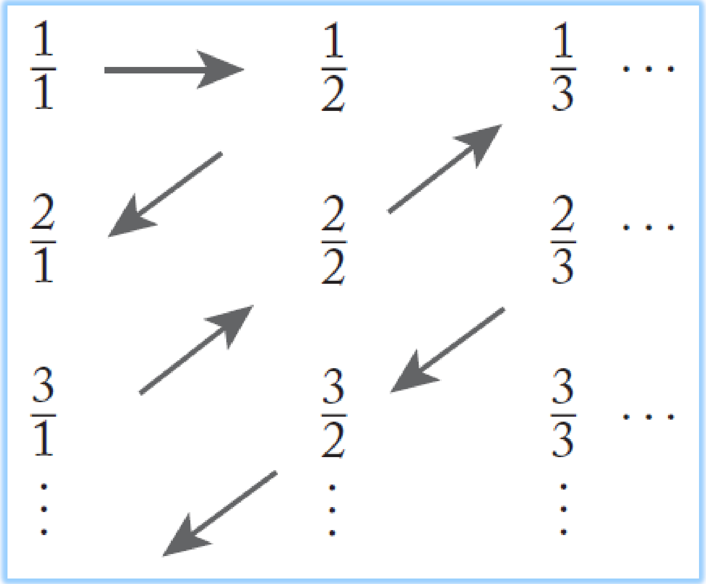
$$\boxed{\delta(q_1, a_2) = (q_2, a_3, L)} \xrightarrow{\text{Encoded}} \boxed{\cdots 1\,0\,1\,1\,0\,1\,1\,0\,1\,1\,1\,0\,1\,0 \cdots}$$

A Universal Turing Machine

- **Some results from set theory**
  - For infinite sets, we distinguish between sets that are countable and sets that are uncountable
  - A set is said to be countable if its elements can be put into a one-to-one correspondence with the positive integers
    - For example, the set of all even integers can be written in the order 0, 2, 4, ...
      - This set is countable
  - A set is countable if we can produce a method by which its elements can be written in some sequence
    - Such a method is called an enumeration procedure
  - Not every set is countable

A Universal Turing Machine

- **Some results from set theory**
  - **Example**
    - Take the set of all quotients of the form p/q, where p and q are positive integers
    - This set is countable by the following enumeration procedure



$$\frac{1}{1}, \frac{1}{2}, \frac{2}{1}, \frac{3}{1}, \frac{2}{2}, \ldots.$$

A Universal Turing Machine

- **Some results from set theory**
  - **Example**

Let $\Sigma = \{a, b, c\}$. We can show that the $S = \Sigma^+$ is countable

$$a, b, c, aa, ab, ac, ba, bb, bc, ca, cb, cc, aaa, \ldots.$$

- **Any Turing machine has a finite encoding as a string on $\{0, 1\}^+$**
  - **The set of all Turing machines, although infinite, is countable**

Linear Bounded Automata

- ***While it is not possible to extend the power of the standard Turing machine by complicating the tape structure, it is possible to limit it by restricting the way in which the tape can be used***
- **We can allow the machine to use only that part of the tape occupied by the input**
  - **Thus, more space is available for long input strings than for short ones**
  - **This generates another class of machines, the linear bounded automata (or lba)**

Linear Bounded Automata

A linear bounded automaton is a nondeterministic Turing machine $M = (Q, \Sigma, \Gamma, \delta, q_0, \square, F)$, as in Definition 10.2, subject to the restriction that Σ must contain two special symbols $[$ and $]$, such that $\delta(q_i, [)$ can contain only elements of the form $(q_j, [, R)$, and $\delta(q_i,])$ can contain only elements of the form $(q_j,], L)$.

- **Example**

  The language

  $$L = \{a^n b^n c^n : n \geq 1\}$$

  is accepted by some linear bounded automaton.

  - **Because its computation does not require space outside the original input**

A Hierarchy of Formal Languages and Automata

Recursive and Recursively Enumerable Languages

A language L is said to be **recursively enumerable** if there exists a Turing machine that accepts it.

- **The definition says nothing about what happens for *w* not in *L***
  - **It may be that the machine halts in a nonfinal state or that it never halts and goes into an infinite loop**

A language L on Σ is said to be **recursive** if there exists a Turing machine M that accepts L and that halts on every w in Σ^+.

The family of recursive languages is a proper subset of the family of recursively enumerable languages.

Recursive and Recursively Enumerable Languages

- There are languages that are not recursively enumerable
- **Example**
  - Let S be an infinite countable set. Then its powerset 2^S is not countable.
    - Let $S = \{s_1, s_2, s_3, \ldots\}$
    - Then any element t of 2^S can be represented by a sequence of 0's and 1's, with a 1 in position i if and only if s_i is in t
    - For example, the set $\{s_2, s_3, s_6\}$ is represented by $01100100\ldots$, while $\{s_1, s_3, s_5, \ldots\}$ is represented by $10101\ldots$

Recursive and Recursively Enumerable Languages

- **Example (Cont.)**
  - Let S be an infinite countable set. Then its powerset 2^S is not countable.
    - **Suppose that 2^S were countable; then its elements could be written in some order, say $t_1, t_2, t_3, \ldots$**
    - **Take the elements in the main diagonal, and complement each entry**
    - **It cannot be t_1, t_2, t_3 or any other entry in the enumeration**

t_1	1	0	0	0	0	$\ldots$
t_2	1	1	0	0	0	$\ldots$
t_3	1	1	0	1	0	$\ldots$
t_4	1	1	0	0	1	$\ldots$
$\vdots$					$\ddots$	

Recursive and Recursively Enumerable Languages

- For any nonempty Σ, there exist languages that are not recursively enumerable.
  - **A language is a subset of Σ^*, and every such subset is a language**
  - **Therefore, the set of all languages is exactly 2^{Σ^*}**
  - **Since Σ^* is infinite, the set of all languages on Σ is not countable (previous example)**
  - **But the set of all Turing machines can be enumerated, so the set of all recursively enumerable languages is countable**

There must be some languages on Σ that are not recursively enumerable