Lecture 17. Closure properties of recursive and r.e. languages

---

Theorem 1 *The class of recursive languages is closed under*

1. *complementation*
2. *union*
3. *concatenation*
4. *Kleene star (proof similar to (3))*
5. *intersection, (proof similar to (2))*

Note: the set of context-free languages is not closed under complementation, but the set of recursive languages is closed under complementation.

If L_1 and L_2 are recursive, then $L_1 \cup L_2$ is recursive.
Proof:

- Let L_1 and L_2 be two recursive languages.
- Then there exists two Turing machines M_1 and M_2 that *decides* M_1 and M_2, respectively.
- Construct a new 2-tape Turing machine M that operates as follows:
  1. Copy the input string w from the first tape to the second tape.
  2. Simulate M_1 on the first tape. If M_1 halts at y, then M halts at y. If M_1 halts at n, proceed to step 3.
  3. Simulate M_2 on the second tape. If M_2 halts at y, then M halts at y. If M_2 halts at n, then M also halts at n.
- Verify that M indeed decides $L_1 \cup L_2$:
  · If $w \in L_1 \cup L_2 \Rightarrow w \in L_1$ or $w \in L_2 \Rightarrow$ (M_1 halts at y) or (M_2 halts at y) $\Rightarrow M$ halts at y.
  · If $w \notin L_1 \cup L_2 \Rightarrow w \notin L_1$ and $w \notin L_2 \Rightarrow$ (M_1 halts at n) and (M_2 halts at n) $\Rightarrow M$ halts at n.

If L_1 and L_2 are recursive, then L_1L_2 is recursive.
Proof:

- Let L_1 and L_2 be two recursive languages.
- Then there exists two Turing machines M_1 and M_2 that *decides* M_1 and M_2, respectively.
- Construct a nondeterministic 2-tape Turing machine which operates as follows:

  1. nondeterministically splits w into two strings x and y such that $w = xy$, then copy y to the second tape (leaving x on the first tape).
  2. Simulate M_1 on the first tape. If M_1 halts at n, then this computation of M rejects w. If M_1 halts at y, then M continues to simulate M_2 on the second tape. If M_2 halts at y, then M accepts w. If M_2 halts at n, then again this computation rejects w.

- Verify that M indeed decides L_1L_2:
  · If $w \in L_1L_2 \Rightarrow$ there exists x and y such that $w = xy$ where $x \in L_1$ and $y \in L_2 \Rightarrow M$ can choose to split $w = xy$, which will allow it to lead to M_1 accepts x and M_2 accepts y, thus M accepts w in that computation, thus $w \in L(M)$.
  · If $w \notin L_1L_2 \Rightarrow$ for all x and y such that $w = xy$ either $x \notin L_1$ or $y \notin L_2 \Rightarrow$ for all x and y, either M_1 rejects x or M_2 rejects y. Thus M rejects w, and $w \notin L(M)$.

Closure properties of r.e. languages

Theorem 2 *The class of r.e. languages is closed under*

1. *union,*
2. *concatenation,*
3. *Kleene star,*
4. *intersection.*

Proof: For concatenation and Kleene star, use nondeterminism similarly as in the proof of recursive languages being closed under concatenation.
For union and intersection, see Question Bank.

Note: The class of r.e. languages is not closed under complementation. (example later).