CSL253 - Theory of Computation

Tutorial 7

Team Members

1. Kartikeya Nainakhwal - 12341090
2. Paritosh Lahre - 12341550
3. Rahul Dev Reddy - 12342390

Question: 2

Let

$$A_{ECFG} = \{\langle G \rangle \mid G \text{ is a CFG and } \varepsilon \in L(G)\}.$$

Show that A_{ECFG} is decidable.

Problem Statement

We define the language

$$A_{ECFG} = \{\langle G \rangle \mid G \text{ is a context-free grammar and } \varepsilon \in L(G)\}.$$

Prove that A_{ECFG} is decidable; that is, there exists a Turing machine that, given any encoding $\langle G \rangle$, halts and correctly decides whether G generates the empty string ε.

Proof Idea

The key observation is that a CFG G generates ε precisely when its start symbol S is "nullable," meaning there exists a derivation $S \Rightarrow_R^* \varepsilon$. We can compute the set of all nullable nonterminals by a simple fixpoint algorithm:

1. Initialize Null with any nonterminal that has a production directly deriving ε.
2. Repeatedly add any nonterminal A if there is a production

$$A \to X_1 X_2 \cdots X_k$$

   such that each X_i is either ε or already in Null.

3. Stop when no new symbols can be added. At termination, $S \in \mathsf{Null}$ iff $\varepsilon \in L(G)$.

Proof

* A_{ECFG} is decidable.

Proof. We construct a deterministic Turing machine M that decides A_{ECFG}. On input $\langle G \rangle$, where $G = (V, \Sigma, R, S)$:

1. Parse the input into (V, Σ, R, S); if malformed, **reject**.
2. Let
$$\text{Null} \leftarrow \{A \in V \mid A \to \varepsilon \in R\}.$$
3. **Repeat** the following until fixpoint:
   a. For each production $A \to X_1 X_2 \cdots X_k$ in R, if $\forall i,\ X_i = \varepsilon$ or $X_i \in \text{Null}$, add A to Null.
4. If $S \in \text{Null}$, **accept**; otherwise, **reject**.

Correctness:

- *Soundness*: If M accepts, then $S \in \text{Null}$, so by construction there is a derivation $S \Rightarrow_R^* \varepsilon$, i.e. $\varepsilon \in L(G)$.
- *Completeness*: If $\varepsilon \in L(G)$, any leftmost derivation shows each nonterminal on the path to ε must be nullable. By induction, our loop adds each such nonterminal to Null, including S, so M accepts.
- *Termination*: Each iteration can only add new symbols to the finite set Null and proceeds until no additions are possible. Since $|V|$ bounds the additions, M halts after at most $|V| + 1$ iterations. All other steps are finite. Thus M always halts.

□

Conclusion

We have exhibited a concrete algorithm that, given any CFG G, computes whether ε is in its language by determining nullable nonterminals. The procedure halts on every input and correctly decides membership in A_{ECFG}. Hence, A_{ECFG} is decidable.