Automata & Formal Languages
Michaelmas Term 2022
Part II of the Mathematical Tripos
University of Cambridge
Prof. Dr. B. Löwe

Union Grammars for Regular Languages

In § 1.7, we defined the *union grammar of G and G'* to be $(\Sigma, V \cup V' \cup \{T\}, P^*, T)$ with a new variable T and $P^* := \{T \to S, T \to S'\} \cup P \cup P'$ and claimed that if G and G' are regular, then so it their union grammar. However, as a very watchful student pointed out, the production rules $T \to S$ and $T \to S'$ are not regular production rules. We had a similar issue with the *concatenation grammar* where the rule $T \to SS'$ was not regular and this was fixed later in Proposition 2.2 with an alternative construction of a regular concatenation grammar.

In order to produce a regular union grammar, an analogous alternative definition is needed. If $G = (\Sigma, V, P, S)$ and $G' = (\Sigma, V', P', S')$, then the *regular union grammar* is defined as

$$H = (\Sigma, V \cup V' \cup \{T\}, P^*, T)$$

with a new variable T and

$$P^* := P \cup P' \cup \{T \to \alpha\,;\, S \to \alpha \in P\} \cup \{T \to \alpha\,;\, S' \to \alpha \in P'\}.$$

With this definition, it is clear that if G and G' are regular, then so is H; similarly, if G and G' are context-free or context-sensitive, then so is H.

Claim. If G and G' are context-sensitive and $V \cap V' = \varnothing$, then $\mathcal{L}(H) = \mathcal{L}(G) \cup \mathcal{L}(G')$.

[The direction "$\supseteq$" is obvious since any derivation $S \xrightarrow{G} w$ or $S' \xrightarrow{G'} w$ can be made into a derivation $T \xrightarrow{H} w$ by exchanging the first rule application rewriting either S or S' by the corresponding rule in P^* rewriting T.

For the other direction, suppose $T \xrightarrow{H} w$. Since H is context-sensitive, all strings occurring in this derivation except for the last one must contain variables (once a string is a word, nothing can be rewritten anymore as every production rule needs a variable to be rewritten). If $T \xrightarrow{H}_1 w$, i.e., the derivation has length one, then it is the result of a rule application of $T \to w$. By definition, either $S \to w \in P$ or $S' \to w \in P'$, so $w \in \mathcal{L}(G) \cup \mathcal{L}(G')$. Otherwise, we have $T \xrightarrow{H}_1 \alpha \xrightarrow{H} w$ with α containing variables. That $T \xrightarrow{H}_1 \alpha$ is either witnessed by some rule $S \to \alpha \in P$ or some rule $S' \to \alpha \in P'$. In the former case, all variables in α are in V; in the latter case, all variables in α are in V'. W.l.o.g., let's assume that we are in the first situation, i.e., all variables in α are in V and $S \to \alpha \in P$. By induction (and the fact that H is context-sensitive, i.e., only variables get rewritten), all variables occurring in the rest of the derivation $\alpha \xrightarrow{H} w$ will also be in V, so all rules applied in the derivation come from P and thus we have $\alpha \xrightarrow{G} w$. But now $S \xrightarrow{G}_1 \alpha \xrightarrow{G} w$, thus $S \xrightarrow{G} w$, and therefore $w \in \mathcal{L}(G) \subseteq \mathcal{L}(G) \cup \mathcal{L}(G')$.]

23 October 2022