How to Demonstrate Metalinearness by Tree-Restricted General Grammars

Martin Havel (ihavelm@fit.vut.cz);
Supervisor: Alexander Meduna

October 30, 2023

Extended Abstract

Demonstrating that some languages are k-linear is important because compilers are relatively simple with k-linear or linear grammars in contrast to general grammars. So, even showing that some nontrivial subset of general grammars are k-linear, brings significant advantages and benefits to many branches of computer science. Therefore, proving that the language generated by a given grammar is k-linear is an essential problem that needs to be studied in depth.

Firstly, this presentation provides essential knowledge to understand the main result, such as general grammars, sentential forms, proving techniques, k-linearness, k-linear grammars, and subsequentially linear grammars. Then, we introduce derivation trees for general grammars and a degree of derivation trees. Within these trees, we define context-dependent pairs of nodes, which corresponds to rewriting two neighboring symbols by a non-context-free rule. The presentation will follow up with introducing a general grammar with a linear core and a slow-branching tree and their relevance to demonstrating metalinearness.

The main result shows that the language generated by a general grammar with the linear core is k-linear if there is a constant k and l such that there is a slow-branching derivation tree of degree k, which satisfies the following:

1. any two nonterminal neighboring paths contain no more than l pairs of context-dependent nodes;

2. out of neighboring paths, any pair of nodes is context-independent.

In the presentation, a simplified step-by-step proof scheme will be demonstrated, so we make it clear how to use it.

Therefore, the presentation provides a take on the construction of a k-linear grammar from a general grammar with a specific condition met, where it is proven that languages generated by these grammars are equivalent. The tool of these capabilities can bring many advantages and fill the gap in the theory of formal languages since this problem is undecidable.