GRAMMAR FORMALISMS VIEWED AS
EVOLVING ALGEBRAS

ABSTRACT. We consider the use of evolving algebra methods of specifying grammars for natural languages. We are especially interested in distributed evolving algebras. We provide the motivation for doing this, and we give a reconstruction of some classic grammar formalisms in directly dynamic terms. Finally, we consider some technical questions arising from the use of direct dynamism in grammar formalisms.

1. Introduction

Formal work in linguistics has both produced and used important mathematical tools. It led to formal language theory, and later developments in that field have found their way back to linguistics. But in addition, ideas originally developed for other applications have been incorporated into linguistic research. This paper considers the use of techniques from the theory of evolving algebras (see Gurevich (1992, 1994)) in the development of syntactic formalisms.

There have been a number of formalisms proposed in the theoretical computer science literature for various kinds of machines which embody distributed behavior and also have a communication facility. Among these are Petri nets (see Petri (1973) and Reisig (1985)), the Chemical Abstract Machine (Berry and Boudol (1992)), the \( \pi \)-calculus (Milner (1992)), and others. Our application of evolving algebras to grammar formalisms may be viewed as another instance of this trend. The framework of evolving algebras is a bit different than the others we mention, mostly because it is geared towards operational semantics and not toward being a general purpose logical formalism.

We suggest below that many grammar frameworks are static formalizations of intuitively dynamic ideas. Part of our motivation is to expose the hidden dynamism: we believe that making it primary might be useful in cross-framework comparisons, and in mathematical and computational linguistics. Further, we show how to render a number of frameworks as evolving algebras. Indeed, the machinery needed to express grammar formalisms is essentially the same as that needed in the operational semantics of programming languages. In this way, we again see a link between the mathematics of grammar formalisms and of generalizations of automata theory.
1.1. Implicit Dynamism in Static Frameworks

In a certain sense, the main point of this paper is a very simple one. We believe that many of the notions found in current and classical syntactic frameworks are prima facie dynamic. At the same time, the mathematical foundations of these frameworks tend to hide this dynamism. There are a number of reasons for this; it was an understandable move in the history of linguistics. However, several lines of research in the past decade have shown ways to reincorporate dynamism in computational settings. So it is possible now to reassess the mathematical models used in grammar formalisms.

Context-free grammars (CFG's) are the main technical tool in a large number of formalized studies in linguistics. We have two main observations on their use. First, the intuitive operations embodied in CF derivations are those of growing a tree (according to rules), and extracting its yield. Now the mathematical definitions of parse tree and yield characterize the crucial data structures implicit in the intuitive ideas, but they are not direct formalizations of their dynamic aspect.

Our second observation is that even if CF grammars and derivations are the backbone of natural language syntax, the “vital organs” are never so simple. Indeed, they are predominantly dynamic. We have in mind concepts like extraction, extraposition, adjunction, advancement, etc., and all types of movement and reconfiguration of tree-like structures. Here, the static formalization of CFG’s conflicts with these intuitively dynamic concepts. It is almost as if the dynamics needed for grammar formalisms is the driving motor of the theory, and CFG’s are just a nice starting point for the formalization of a number of ideas.

We believe that the great majority, if not all, of the models used in syntax before, say, 1980, were essentially dynamic. Indeed, the first formalized, completely declarative linguistic framework was arguably Arc Pair Grammar (Johnson and Postal (1980)). (Today, the field of linguistics is split – the Chomskyan school appears dynamic, but more declarative formulations of grammar, including those based on some variety of feature logic, have become popular.) Our work should provide a way to make procedural frameworks respectable to those wary of them.

1.2. Goals of this Work

We propose a framework for defining and studying grammar derivations as dynamic or evolving phenomena. Specifically, we adapt the methodology of evolving algebras from programming languages to mathematical