Abstract. We provide a characterization of the local sets (sets of trees generated by CFGs) in terms of definability in a restricted logical language, which we contrast with a similar characterization of the recognizable sets (sets of trees accepted by finite-state tree automata). In a strong sense, the distinction between these two captures abstractly the distinction between ordinary CFGs and those in which labels are finitely extended with additional features (as in GPSG). In terms of descriptive complexity, the contrast is quite profound—while the recognizable sets are characterized by a monadic second-order language, the local sets are characterized by a severely restricted modal language. In the domain of strings, there is an analogous contrast between the regular languages and the strict 2-locally testable languages, a weak subclass of the star-free sets.

1 Introduction

In the early sixties, in the context of exploring decidability of properties of sequences, Büchi [3] and Elgot [6] established a descriptive characterization of the regular sets in terms of definability in the monadic second-order theory of the natural numbers with successor—the theory that is now known as S1S. The idea is that if we interpret an alphabet \( \Sigma \) as a set of monadic predicate variables, then we can interpret non-empty sequences drawn from \( \Sigma \), i.e., possibly infinite non-empty strings in \( \Sigma \), as assignments associating each variable \( a \in E \) with a subset of \( N \): namely, the set of positions in the string at which \( a \) occurs. The regular sets are, under this interpretation, the sets of strings that are satisfying assignments for some formula \( \varphi(\Sigma) \) in the language of S1S.

To be precise, let \( N_1 = \langle N, 0, \leq, s \rangle \) the structure of the natural numbers with zero, the usual ordering, and successor.\(^1\) The language of S1S includes the signature of \( N_1 \), a set of variables ranging over \( N \) and another ranging over arbitrary subsets of \( N \), quantification over both sorts of variables, and the usual logical connectives. S1S is the set of all sentences in this language that are true in \( N_1 \). The set of sequences defined by a formula \( \varphi(\Sigma) \) with free-variables among those in \( \Sigma \) are just those generated by the assignments that make \( \varphi(\Sigma) \) true in \( N_1 \).

Büchi and Elgot were concerned with infinite sequences for which Büchi introduced a class of finite automata over infinite strings that are now referred to

\(^1\) The ordering is actually inessential since it is monadic second-order definable from successor.
as *Büchi automata*. Büchi’s central result is that S1S is decidable. He establishes this by showing that, for any S1S formula, there is a Büchi automaton that accepts exactly the set of its satisfying assignments, and that it is decidable if the set accepted by such an automaton is empty. Then \((\exists X)[\varphi(X)] \in S1S\) if and only if the set accepted by the automaton for \(\varphi(X)\) is non-empty.\(^2\) As finiteness is definable in S1S, one can restrict to finite strings by restricting the sets assigned to the variables in \(\Sigma\) to be finite. There is an additional complication in that one must deal with the empty string, but this is inconsequential. Under these circumstances Büchi’s automata become simple finite state automata. Thus any subset of \(\Sigma^*\) that is definable in S1S is regular and, since the converse is easy to show, definability in S1S characterizes the regular languages.

This result provides a powerful tool for establishing language-theoretic complexity results that is particularly attractive in the realm of constraint-based approaches to syntax: constraints or principles that are definable in S1S can define only regular languages. The utility of the result, of course, is limited by the weakness of the class it characterizes; the regular languages are not particularly interesting from a linguistic point of view. The obvious step, then, is to extend the result to larger language-theoretic complexity classes.

Intuitively, one can approach capturing the context-free languages in a manner similar to Büchi’s characterization of the regular languages by noting that the characteristic operation in generating regular languages is concatenation of strings while the characteristic operation in generating CFLs is substitution in strings, and that substitution in strings is analogous to concatenation of trees in the following sense: substitution of \(\gamma\) for \(X\) in a sentential form \(\alpha X \beta\) corresponds to concatenating the derivation tree for \(\alpha X \beta\) with a local tree (a tree of depth one) at \(X\) expanding \(X\) to \(\gamma\). Now, one can interpret trees as a generalization of the natural numbers with successor in which there are multiple successor functions, i.e., if we take \(N_1 = (\mathbb{N}, 0, \leq, s)\) to be the complete unary branching tree, the complete \(n\)-branching tree is \(N_n = (T_n, \epsilon, s^*, \leq, r_i)_{i \leq n}\) where:

- \(T_n\) is the complete \(n\)-branching tree domain (i.e., \(\{0, \ldots, n-1\}^*\)).
- \(\epsilon\) is the root of \(T_n\).
- \(r_i\) is the \(i\)th successor function: \(r_i(w) = w \cdot i\).
- \(\prec^*\) is reflexive domination: \(w \prec^* w \cdot v\) for all \(w, v \in T_n\).
- \(\preceq^*\) is the lexicographic order on \(T_n\): \(w \preceq^* v\) iff \(w \prec^* v\) or there is some \(u \in T_n\) and \(i < j\) such that \(u \cdot i \prec^* w\) and \(u \cdot j \prec^* v\).\(^3\)

Rabin [13] has explored SnS in a manner similar to Büchi’s exploration of S1S, including defining a type of finite-state automaton over infinite trees, and a similar, but much more difficult, proof of decidability of SnS for all \(n \leq \omega\). Again in this context, if we restrict our attention to the finite structures we find ourselves in more familiar territory. In the finite case, Rabin automata reduce to ordinary finite-state tree automata [8]. One gets, then, that sets of finite trees

\(^2\) Or, more directly but less intuitively clear, a sentence \(\varphi\) is in S1S iff the automaton for \(\varphi\) accepts; since \(\Sigma\) is empty, acceptance is independent of the input.

\(^3\) Again, \(\prec^*\) is definable except in \(N_\omega\)—the \(\omega\)-branching tree—as is \(\preceq^*\). In \(N_1\) \(\prec^* = \preceq^* = \leq\).