

15 Prefix-Recognizable Graphs and Monadic Logic

Martin Leucker*

Department of Computer Systems
Uppsala University

15.1 Introduction

In 1969, Rabin [148] showed that the monadic second-order theory (MSO-theory) of infinite binary trees is decidable (see Chapter 12 of this volume or [183]). Ever since, it has been an interesting goal to extend this result to other classes of objects.

Muller and Schupp [135] showed that the class of *pushdown graphs* has a decidable MSO-theory. This class is obtained by considering the configuration graphs of pushdown machines. The result was later extended to the class of *regular graphs* introduced by Courcelle [42], which are defined as solutions of graph-grammar equations.

Prefix-recognizable graphs were introduced by Caucal in [28]. They extend the pushdown graphs of Muller and Schupp and the regular graphs of Courcelle. Originally, Caucal introduced this class of graphs via transformations on the complete infinite binary tree. The decidability result of their MSO-theory was obtained by showing that these transformations are definable by MSO-formulas. Hereby, the decidability result of the MSO-theory of trees was transferred to the class of prefix-recognizable graphs. The approach can also be understood as interpreting prefix-recognizable graphs in the infinite binary tree by means of MSO-formulas. Barthelmann [6] and Blumensath [12] showed independently that Caucal's class of graphs coincides with the class of graphs MSO-interpretable in the infinite binary tree. In simple words, prefix-recognizable graphs provide a decidability proof of their MSO-theory via MSO-interpretations in the infinite binary tree.

The aim of this chapter is to present prefix-recognizable graphs and to show several of their representations. In contrast to Caucal's original outline, we start with graphs that are MSO-interpretable in the binary tree. In this way, we obtain a natural class of graphs which trivially have a decidable MSO-theory (see Section 15.3). We then provide several representations of these graphs in Section 15.3 and Section 15.5. We learn that prefix-recognizable graphs can be represented by means of prefix-transition systems, whose prefixes form regular languages, justifying the name of this class. Furthermore, we introduce Caucal's transformations on the binary tree and show that they induce the same class of graphs.

* Supported by European Research Training Network "Games".

Although the class of prefix-recognizable graphs is the largest *natural* class proving a decidability via interpretation in the binary tree, it should be mentioned that there are graphs which are not prefix-recognizable but have a decidable MSO-theory (see [12]). A different natural class of structures having a decidable MSO-theory is presented in Chapter 16.

This chapter is organized as follows. In Section 15.2, we fix our notation and introduce basic concepts. Section 15.3 introduces prefix-recognizable graphs as graphs which are MSO-interpretable in the infinite binary tree and provides a first representation. In Section 15.4 we present transformations on graphs and show some of their properties. Applying all these transformations on the binary tree will yield the same class of graphs. This is shown in Section 15.5. A characterization in terms of pushdown graphs is given in Section 15.6. We conclude by summarizing our results and by giving further representations.

15.2 Preliminaries

We denote alphabets by Σ, Γ, \dots , and N . The most important alphabet considered in this chapter is the **binary alphabet** consisting of 0 and 1. It is denoted by $\mathbb{B} = \{0, 1\}$. As usual, a **language** over Σ is a subset of Σ^* , the set of finite sequences of elements of Σ . The elements of Σ^* are called **words**. The empty word is denoted by ε . Σ^+ denotes $\Sigma^* \setminus \{\varepsilon\}$. The class of **regular languages** over Σ is denoted by $\text{REG}(\Sigma^*)$. For two sets of words U and V , we denote by UV their **language product**, i.e., $UV = \{uv \mid u \in U, v \in V\}$.

A **tree** over an alphabet N is a structure $\mathfrak{T} = (T, (\sigma_a)_{a \in N})$. Here, $T \subseteq N^*$ is prefix-closed and is called the **domain** of \mathfrak{T} . The a -successor relation σ_a contains all pairs (x, xa) for $xa \in T$. The **complete tree** over N is $\mathfrak{T}_N := (N^*, (\sigma_a)_{a \in N})$. It is sometimes convenient to regard trees as partial orders (T, \preceq, \sqcap) , where \preceq is the prefix-ordering and $x \sqcap y$ denotes the longest common prefix of x and y . Further, we identify a prefix-closed set $T \subseteq N^*$ with the tree $(T, (\sigma_a)_{a \in N})$. A **Σ -labelled tree** (Σ -tree for short) is either represented as a structure $(T, (\sigma_a)_{a \in N}, (P_a)_{a \in \Sigma})$ with $P_a \subseteq T$ or simply as a mapping $T \rightarrow \Sigma$. Finally, a **regular tree** is a tree with only finitely many subtrees up to isomorphism. In the following, by tree we will usually mean a complete infinite tree.

Figure 15.1 shows part of the infinite binary tree. Note that node 0 has successors 00 and 01. Hence, node s' is a descendant of s iff $s' = su$ for an appropriate word u . Sets of successors might therefore be represented by sets of suffixes. Originally, sets of descendants were identified by prefixes, yielding the notion of *prefix-recognizable graphs*. We follow the suffix approach because it simplifies our notation. It is clear that everything presented in this chapter can be turned into a “prefix” version by a simple “reversal” operator.

The kind of graphs we are going to consider are edge-labelled directed graphs. As for trees, the vertices will usually be words over some alphabet N , and the edge labels are from some alphabet Σ . Such a graph is also called **Σ -graph**. The edge set is partitioned into sets E_a collecting the edges labelled a . Thus

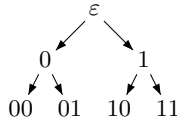


Fig. 15.1. A part of the binary tree

graphs will be represented in the form $\mathfrak{G} = (V, (E_a)_{a \in \Sigma})$. For convenience we allow graphs to be represented also in the form $\mathfrak{G} = (V, E)$ for $V \subseteq N^*$ and $E \subseteq N^* \times \Sigma \times N^*$. Sometimes, we simply use a ternary relation $E \subseteq N^* \times \Sigma \times N^*$ for a graph \mathfrak{G} , in which case we assume $V_{\mathfrak{G}}$ (the set of **nodes of \mathfrak{G}**) to be implicitly defined by $V_{\mathfrak{G}} = \{s \mid \exists a \in \Sigma, \exists t \in N^*. (s, a, t) \in E \text{ or } (t, a, s) \in E\}$. It is obvious that our notion of graphs subsumes that of trees.

Another key feature of our notion of graphs is that their nodes can be associated with words over some alphabet Σ . Hence, the nodes of graphs constitute languages. It is a traditional task to deal with finite representations of infinite languages by means of automata. Taking languages as the domain for our node sets, we provide the framework of automata theory for defining, characterizing, and modifying the corresponding graphs.

Let $\mathfrak{G} = (V, (E_a)_{a \in \Sigma})$ be a graph. An edge $(s, t) \in E_a$ is denoted by $s \xrightarrow[\mathfrak{G}]{a} t$, or, if \mathfrak{G} is fixed, by $s \xrightarrow{a} t$. A **path** from s to t in \mathfrak{G} via a word $u = a_1 \dots a_k \in \Sigma^*$ is a sequence

$$s = p_1 \xrightarrow[\mathfrak{G}]{a_1} \dots \xrightarrow[\mathfrak{G}]{a_k} p_{k+1} = t$$

for $s, t \in V$, and appropriate nodes $p_i \in V$. We write $s \xrightarrow[\mathfrak{G}]{u} t$ iff there is path from s to t via u . Again, we may omit the subscript \mathfrak{G} if it is clear from the context. We write $s \xrightarrow{L} t$ to denote that there is path from s to t via a word u which is in L . A **root** of a graph is a node from which all other nodes are reachable, i.e., $r \in V$ is a root iff for all $s \in V$ there is a $u \in \Sigma^*$ such that there is path from r to s via u .

Given a sequence of edges, its sequence of labels is a word over the alphabet Σ . Given two nodes s and t of a graph \mathfrak{G} , we define the language $L(\mathfrak{G}, s, t)$ to be the union of all words which are obtained on paths from s to t in \mathfrak{G} in the way described above. The union of $L(\mathfrak{G}, s, t)$ for arbitrary nodes s and t of \mathfrak{G} is abbreviated by $L(\mathfrak{G})$.

A letter a can be associated with the set of a -labelled edges $\{p \xrightarrow{a} q\}$, while a word $w = a_1 \dots a_n$ over Σ can be associated with the w -paths from a node p to a node q , $p = p_0 \xrightarrow{a_1} \dots \xrightarrow{a_n} p_n = q$.

Given sets of words W, U , and V we denote by $W(U \xrightarrow{a} V)$ the set of edges $\{wu \xrightarrow{a} wv \mid w \in W, u \in U, v \in V\}$. In a similar manner, we define $U \xrightarrow{a} V$. A graph \mathfrak{G} is called **recognizable** iff there are a natural number n and $a_1, \dots, a_n \in \Sigma$, $U_1, V_1, \dots, U_n, V_n \in \text{REG}(\Sigma^*)$ such that $\mathfrak{G} = U_1 \xrightarrow{a_1} V_1 \cup \dots \cup U_n \xrightarrow{a_n} V_n$.

Let us recall the automata theoretic notations from Chapter 1. We denote finite automata over words by tuples $\mathcal{A} = (Q, \Sigma, \Delta, q_0, F)$ with a set of states

Q , alphabet Σ , transition relation Δ , initial state q_0 , and acceptance condition F . Sometimes, if the automaton is deterministic, Δ is replaced by a function δ . The language accepted by \mathcal{A} is denoted by $L(\mathcal{A})$.

The power set of a set Σ is denoted by $\mathcal{P}(\Sigma)$.

Let us recall some basic definitions regarding monadic second-order logic. MSO-logic extends first-order logic FO by quantification over sets. First-order variables are usually denoted by x, y, \dots and second-order variables by X, Y, \dots, X_1, \dots . We write $\varphi(x, y)$ to denote that φ has free variables among x and y . The **theory** of a structure comprises all formulas that hold in the structure. For a graph \mathfrak{G} , we denote by $\text{MTh}(\mathfrak{G})$ its theory. For a thorough introduction to MSO-logic we refer to Chapter 12 of this volume.

15.3 Prefix-Recognizable Graphs

Given a structure \mathfrak{B} , an MSO-formula $\varphi(x)$ with a free first order variable x induces the set $B = \varphi^{\mathfrak{B}}(x) = \{b \mid \mathfrak{B} \models \varphi(b)\}$. We can specify a graph $\mathfrak{G} = (V, E)$ by providing MSO-formulas $\varphi(x)$ and $\psi(x, y)$ such that $V = \varphi^{\mathfrak{B}}(x)$ and $E = \psi^{\mathfrak{B}}(x, y)$ (defined analogously). In this case we say that \mathfrak{G} is MSO-interpretable in \mathfrak{B} via the formulas φ, ψ . Interpretations are a general tool for obtaining classes of finitely presented structures with sets of desired properties.

In this section we introduce a class of graphs via MSO-interpretations in the infinite binary tree. Since the latter has a decidable MSO-theory, we obtain a natural class of graphs with a decidable MSO-theory. Furthermore, we give a representation of these graphs in terms of prefix transition graphs whose prefixes form regular sets of words. This justifies the name *prefix-recognizable graphs*.

Let us make the notion of an MSO-interpretation precise:

Definition 15.1. Let $\mathfrak{A} = (A, R_1, \dots, R_n)$ and \mathfrak{B} be relational structures. A (one-dimensional) **MSO-interpretation** of \mathfrak{A} in \mathfrak{B} is a sequence

$$I = (\delta(x), \varepsilon(x, y), \varphi_{R_1}(\bar{x}), \dots, \varphi_{R_n}(\bar{x}))$$

of MSO-formulas such that

$$\mathfrak{A} \cong (\delta^{\mathfrak{B}}(x), \varphi_{R_1}^{\mathfrak{B}}(\bar{x}), \dots, \varphi_{R_n}^{\mathfrak{B}}(\bar{x})) / \varepsilon^{\mathfrak{B}}(x, y)$$

To make the previous structure well-defined, we require $\varepsilon^{\mathfrak{B}}$ to be a congruence of the structure $(\delta^{\mathfrak{B}}(x), \varphi_{R_1}^{\mathfrak{B}}(\bar{x}), \dots, \varphi_{R_n}^{\mathfrak{B}}(\bar{x}))$.

We write $I : \mathfrak{A} \leq_{\text{MSO}} \mathfrak{B}$ if I is an MSO-interpretation of \mathfrak{A} in \mathfrak{B} . Since \mathfrak{A} is uniquely determined by \mathfrak{B} and I , we can regard I as a functor and denote \mathfrak{A} by $I(\mathfrak{B})$. The **coordinate map** from $\delta^{\mathfrak{B}}(x)$ to A , the universe of \mathfrak{A} , is also denoted by I . We call I **injective** if the coordinate map is injective.

If I is clear from the context, or if we want to express that there is an interpretation of \mathfrak{A} in \mathfrak{B} , we simply write $\mathfrak{A} \leq_{\text{MSO}} \mathfrak{B}$. In the latter case, we also say that \mathfrak{A} is MSO-interpretable in \mathfrak{B} .

Example 15.2. Words are MSO-interpretable in the binary tree. Intuitively, an infinite word can be obtained in the infinite binary tree by considering a single branch. First, observe that $\text{Root}(y) = \neg\exists x S_1xy$ identifies the root of a tree. Let us take the path obtained by always considering the right successor. Then, the universe of a word is the minimal set of nodes that contains a root and all right successors. Thus we define

$$\delta(x) = \forall U(\exists yUy \wedge \text{Root}(y) \wedge \forall p\forall q(Up \wedge S_1pq \rightarrow Uq) \rightarrow Ux)$$

The successor relation is simply defined by $\varphi_{S_0}(x, y) = S_1xy$ and every set of labels P_a can be defined by $\varphi_{P_a}(x) = P_ax$. It is now easy to see that $I = (\delta(x), \varepsilon(x, y), \varphi_{S_0}, (\varphi_{P_a})_{a \in \Sigma})$ is an MSO-interpretation of the word \mathfrak{A} in \mathfrak{B} where ε is assumed to express the identity relation of $\delta^{\mathfrak{A}}$.

Exercise 15.1. A structure $\mathfrak{A} = (A, <^{\mathfrak{A}})$ is called a *dense open order* if $<^{\mathfrak{A}}$ is a total order on A , if for all $x \in A$ there are $y, z \in A$ such that $y <^{\mathfrak{A}} x <^{\mathfrak{A}} z$, and if for all $x, y \in A$ such that $x <^{\mathfrak{A}} y$ there is a $z \in A$ such that $x <^{\mathfrak{A}} z <^{\mathfrak{A}} y$. Show that a dense open order can be interpreted in the infinite binary tree.

Theorem 15.3. *If $\mathfrak{A} \leq_{\text{MSO}} \mathfrak{B}$ and \mathfrak{B} has a decidable MSO-theory then \mathfrak{A} has a decidable MSO-theory.*

Proof. We give a sketch of the proof. The details are left as an exercise for the reader. Let $I = (\delta(x), \varepsilon(x, y), \varphi_{R_1}(\bar{x}), \dots, \varphi_{R_n}(\bar{x}))$ be an MSO-interpretation of \mathfrak{A} in \mathfrak{B} . Consider a formula φ . Let φ' be obtained from φ in the following way: Replace every relational symbol R in φ by its defining formula φ_R . Furthermore, relativize every quantifier to $\delta(x)$, i.e. substitute $\exists x\varphi$ by $\exists x(\delta(x) \wedge \varphi)$ and $\forall x\varphi$ by $\forall x(\delta(x) \rightarrow \varphi)$. Now it is easy to see that $\mathfrak{A} \models \varphi$ iff $\mathfrak{B} \models \varphi'$.

Since we are interested in interpreting graphs with labelled edges in structures, we deal with interpretations of the form $I = (\delta(x), \varepsilon(x, y), (\varphi_{R_a}(x, y))_{a \in \Sigma})$.

The decidability of the monadic second-order theory of $\mathfrak{T}_{\mathbb{B}}$ was established by Rabin in [148]. Thus, if we consider MSO-interpretations in the binary tree we get structures with a decidable MSO-theory.

Corollary 15.4. *Every graph which is MSO-interpretable in the infinite binary tree $\mathfrak{T}_{\mathbb{B}}$ has a decidable monadic second-order theory.*

Let us now give a representation of the graphs which are MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$ in terms of prefix-transition graphs having regular prefixes.

Definition 15.5. Let Σ be an alphabet. A graph $\mathfrak{G} = (V, (E_a)_{a \in \Sigma})$ is called **prefix-recognizable** iff it is isomorphic to a graph of the form

$$\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$$

for some $n \geq 0, a_1, \dots, a_n \in \Sigma$ and languages $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(\mathbb{B}^*)$. The class of prefix-recognizable graphs with edge labels among Σ is denoted by $\text{PRG}(\Sigma)$ or PRG if Σ is fixed.

We will show in Section 15.5 that we can choose an arbitrary alphabet with at least two elements instead of \mathbb{B} .

Example 15.6. Let us consider the graph with edge labels a and b given by $\mathbb{B}^* ((\varepsilon \xrightarrow{a} \mathbb{B}) \cup \mathbb{B}^* . (\mathbb{B}^+ \xrightarrow{b} \varepsilon))$. It is depicted in Figure 15.2. Note that this graph is (A^*, R_a, R_b) is isomorphic to $(\omega, \text{succ}, >)$, where succ is the successor relation on the natural numbers.

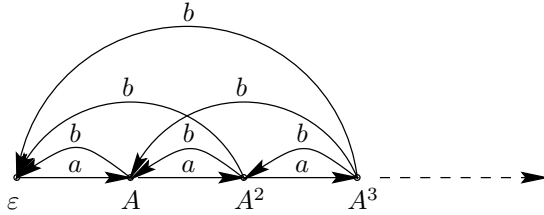


Fig. 15.2. A prefix-recognizable graph with infinite out-degree

This example shows that prefix-recognizable graphs may have nodes with infinite out-degree. The class of prefix-recognizable graphs is a strict extension of the class of regular graphs, since the latter have only a finite out-degree [42].

Every prefix-recognizable graph can be represented by a finite collection of **prefix-recognizable rewrite rules** $w.u \xrightarrow{a} v$ where $w, u,$ and v are regular expressions. This way of representing prefix-recognizable graphs will be employed in Chapter 17.

Let us proceed to show that prefix-recognizable graphs coincide with graphs that are MSO-definable in $\mathfrak{T}_{\mathbb{B}}$. In our constructions, we need to code tuples of sets as labelled trees.

Definition 15.7. For sets $X_0, \dots, X_{n-1} \subseteq \mathbb{B}^*$, abbreviated by \bar{X} , denote by $T_{\bar{X}}$ the \mathbb{B}^n -labelled binary tree such that the i -th component of the label $T_{\bar{X}}(y)$ for a node y is 1 iff $y \in X_i$. Singletons $X_i = \{x_i\}$ ($i = 0, \dots, n - 1$) are also abbreviated by \bar{x} .

Furthermore, we employ Rabin’s tree theorem which gives the relation between tree automata and MSO-logic: (see also Chapter 12 and [179])

Theorem 15.8. For each $\varphi(\bar{X}, \bar{x}) \in \text{MSO}$ there is a tree-automaton \mathcal{A} such that $L(\mathcal{A}) = \{T_{\bar{X}\bar{x}} \mid \mathfrak{T}_{\mathbb{B}} \models \varphi(\bar{X}, \bar{x})\}$.

Let us establish the representation first for injective interpretations.

Proposition 15.9. Let \mathfrak{G} be a graph which is MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$ via an injective interpretation. Then \mathfrak{G} is isomorphic to $\bigcup_{i=1}^n W_i (U_i \xrightarrow{a_i} V_i)$ for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(\mathbb{B}^*)$.

Proof. Let $I : \mathfrak{G} \leq_{\text{MSO}} \mathfrak{F}_{\mathbb{B}}$ be an injective MSO-interpretation of \mathfrak{G} in $\mathfrak{F}_{\mathbb{B}}$. Note that $\mathfrak{G} = (V, (E_a)_{a \in \Sigma})$ and every edge relation E is defined by a formula $\varphi(x, y)$. We have to show that every such edge relation E can be written as a finite union of $W(U \rightarrow V)$ where U, V , and W are regular. Let $\mathcal{A} = (Q, \mathbb{B}, \Delta, q_0, \Omega)$ be the tree-automaton associated with φ with respect to $\mathfrak{F}_{\mathbb{B}}$ (cf. Chapter 12). Thus $L(\mathcal{A}) = \{T_{xy} \mid \mathfrak{F}_{\mathbb{B}} \models \varphi(x, y)\}$. Note that every $T_{xy} \in L(\mathcal{A})$ is labelled by tuples in \mathbb{B}^2 . Nearly all nodes are labelled by $[0, 0]$, except the node representing x , in which the first component of the tuple is 1 and the node representing y , in which the second component of the tuple is 1. Figure 15.3 shows a typical situation. Observe, that each pair x and y can be written as a wu and wv in the way described in the introductory section.

We construct languages U_q, V_q , and W_q such that $u \in U_q, v \in V_q$, and $w \in W_q$ if and only if there is an accepting run of \mathcal{A} on $T_{\{wu\}\{wv\}}$ where the node w is labelled by q . Thus, $E = \bigcup_{q \in Q} W_q(U_q \rightarrow V_q)$. We show that every U_q, V_q, W_q is regular, and since Q is finite, we are done.

Let $Q_0 \subseteq Q$ be the subset of states from which \mathcal{A} accepts the tree labelled by $[0, 0]$ everywhere. Note that this set is computable for \mathcal{A} . Obviously, wu or wv can only occur in a subtree not labelled by $[0, 0]$ everywhere. Hence, a node labelled by the state q from which wu and wv are reachable is in this subtree. Hence, we let W_q be the language recognized by the automaton $(Q, \mathbb{B}, \Delta_{W_q}, q_0, \{q\})$ where

$$\Delta_{W_q} := \{ (p, 0, p') \mid (p, [0, 0], p', p_0) \in \Delta, p_0 \in Q_0 \} \cup \{ (p, 1, p') \mid (p, [0, 0], p_0, p') \in \Delta, p_0 \in Q_0 \}$$

If the desired state q is reached, we have to look for a node labelled by $[1, _]$ for an element of U_q and for a node labelled by $[_, 1]$ for an element of V_q . Hence, we let $U_q := L((Q \cup \{q_f\}, \mathbb{B}, \Delta_{U_q}, q, \{q_f\}))$ where

$$\Delta_{U_q} := \{ (p, 0, p') \mid (p, [0, c], p', p_0) \in \Delta, p_0 \in Q_0, c \in \mathbb{B} \} \cup \{ (p, 1, p') \mid (p, [0, c], p_0, p') \in \Delta, p_0 \in Q_0, c \in \mathbb{B} \} \cup \{ (p, c, q_f) \mid (p, [1, d], p_0, p'_0) \in \Delta, p_0, p'_0 \in Q_0, c, d \in \mathbb{B} \}$$

V_q is defined similar to U_q , only the tuples (labels) are switched.

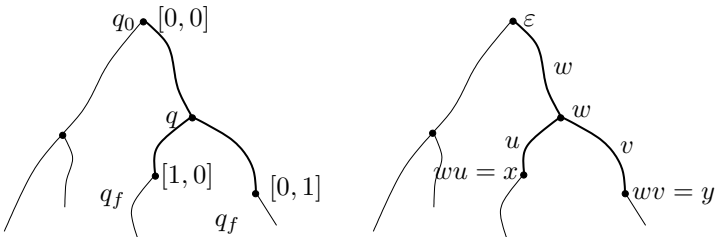


Fig. 15.3. A run of the tree automaton

Let us verify that it suffices to consider injective MSO-interpretations. Hence, we may assume that the equivalence classes with respect to $\varepsilon^{\mathfrak{B}}$ are singletons. Let us first show the following lemma:

Lemma 15.10. *Let $D \subseteq \mathbb{B}^*$ be regular and $E \subseteq D \times D$ an equivalence relation which is prefix-recognizable.¹ There is a regular language $D' \subseteq D$ such that D' contains exactly one element of each E -class.*

Proof. Denote the E -class of x by $[x]$, define $p_{[x]} := \inf_{\preceq} [x]$ and $s_x := (p_{[x]})^{-1}x$. Let $\varphi_p(x, y)$ be an MSO-definition of the function $x \mapsto p_{[x]}$. Finally, let s be the number of states of the tree automaton associated with E . We claim that each class $[x]$ has an element of length less than $|p_{[x]}| + s$. Thus, one can define

$$D' := \{x \in D \mid s_x \leq s_y \text{ for all } y \in [x]\}$$

where \leq is the lexicographic ordering which is definable since the length of the words is bounded so that we only need to consider finitely many cases.

To prove the claim, choose $x_0, x_1 \in [x]$ such that $x_0 \sqcap x_1 = p_{[x]}$. Since $(x_0, x_1) \in E$ there are regular languages U, V , and W such that $x_0 = wu, x_1 = wv$ for $u \in U, v \in V$, and $w \in W$ with $w \preceq p_{[x]}$. If $|wu| \geq |p_{[x]}| + s$ then, by a pumping argument, there exists some $u' \in U$ such that $|p_{[x]}| \leq |wu'| \leq |p_{[x]}| + s$. Hence, $(wu', x_1) \in E$ is an element of the desired length.

Let us return to $\varepsilon^{\mathfrak{B}}$. Since it is a binary relation, it may be understood as the edge relation of a graph. By Proposition 15.9, this is prefix-recognizable, and by the previous lemma, there is a regular set D' which contains for every equivalence class with respect to $\varepsilon^{\mathfrak{B}}$ a single element. Since D' is regular, there is an MSO-formula $\delta'(x)$ defining D' . Hence, if $\mathfrak{A} \cong (\delta^{\mathfrak{B}}(x), \varphi_{R_1}^{\mathfrak{B}}(\bar{x}), \dots, \varphi_{R_n}^{\mathfrak{B}}(\bar{x})) / \varepsilon^{\mathfrak{B}}(x, y)$ then \mathfrak{A} is also isomorphic to $(\delta'^{\mathfrak{B}}(x), \varphi_{R_1}^{\mathfrak{B}}(\bar{x}), \dots, \varphi_{R_n}^{\mathfrak{B}}(\bar{x}))$. Thus, the following corollary holds.

Corollary 15.11.

- (1) PRG is closed under prefix-recognizable congruences.
- (2) Each graph MSO-interpretable in the binary tree has an injective MSO-interpretation in $\mathfrak{T}_{\mathbb{B}}$.

Let us summarize the previous results:

Lemma 15.12. *Suppose \mathfrak{G} is MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$. Then*

$$\mathfrak{G} \text{ is isomorphic to } \bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$$

for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(\mathbb{B}^*)$.

It is easy to see that also the converse holds.

¹ in the sense of E considered as a set of edges

Lemma 15.13. *Let \mathfrak{G} be a graph isomorphic to*

$$\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$$

for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(\mathbb{B}^*)$. Then \mathfrak{G} is MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$.

Proof. We have to show that for each $a \in \{a_1, \dots, a_n\}$ there is a formula $\varphi_{R_a}(x, y)$ interpreting the a -edges in $\mathfrak{T}_{\mathbb{B}}$. Clearly, the prefix-ordering \preceq on binary strings is MSO-definable in $\mathfrak{T}_{\mathbb{B}}$. Further, for each regular language $L \subseteq \mathbb{B}^*$ there exists an MSO-formula $\varphi_L(u, v)$ stating the $u \preceq v$ and the labelling of the path from u to v in $\mathfrak{T}_{\mathbb{B}}$ is in L . The latter can be expressed by the formula $\text{Path}_L(x, y)$ that can be defined inductively on L by:

$$\begin{aligned} \text{Path}_{\emptyset}(x, y) &= \exists X(x \in X \wedge \neg(x \in X)) \quad \text{“false”} \\ \text{Path}_{\{b\}}(x, y) &= S_bxy \\ \text{Path}_{L+M}(x, y) &= \text{Path}_L(x, y) \vee \text{Path}_M(x, y) \\ \text{Path}_{L.M}(x, y) &= \exists z(\text{Path}_L(x, z) \wedge \text{Path}_M(z, y)) \\ \text{Path}_{L^*}(x, y) &= \forall X((x \in X \wedge \\ &\quad \forall p \forall q((p \in X \wedge \text{Path}_L(p, q)) \rightarrow q \in X)) \rightarrow y \in X) \end{aligned}$$

We now see that, $\bigcup_{\{i|a=a_i\}} W_i(U_i \xrightarrow{a_i} V_i)$ can be defined by

$$\varphi_{R_a}(x, y) = \bigvee_{\{i|a=a_i\}} \exists z(\varphi_{W_i}(\varepsilon, z) \wedge \varphi_{U_i}(z, x) \wedge \varphi_{V_i}(z, y)).$$

Combining the previous two lemmas we get

Theorem 15.14. *A graph \mathfrak{G} is MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$ iff it is isomorphic to*

$$\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$$

for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(\mathbb{B}^*)$.

In other words, a graph is MSO-interpretable in $\mathfrak{T}_{\mathbb{B}}$ iff it is prefix-recognizable.

Caucal introduced prefix-recognizable graphs employing transformations on the binary tree instead of MSO-interpretations. We will redevelop his approach in the next section, obtaining further representations of prefix-recognizable graphs.

15.4 Transformations on Graphs

In this section, we introduce several transformations on the complete infinite binary tree. For the first two transformations, we will prove that they are definable within MSO, giving rise to MSO-interpretations of graphs in $\mathfrak{T}_{\mathbb{B}}$. In other words, we obtain prefix-recognizable graphs by employing our transformations.

We will employ these transformations in the next section to obtain further representations of prefix-recognizable graphs.

The idea of the first transformation is to collapse paths within a given graph to a single edge with a new label in the new graph. To be able to deal with inverse edges of a graph, we introduce the notion of an inverse alphabet.

Definition 15.15. Let Σ be an alphabet. The **inverse alphabet** of Σ is the set $\bar{\Sigma} := \{\bar{a} \mid a \in \Sigma\}$ which is a disjoint copy of Σ . The **extended alphabet** of Σ is the union of Σ and its inverse alphabet and is denoted by $\hat{\Sigma}$.

Words over the extended alphabet of Σ may correspond to paths with inverse edges. For example, the word $\bar{a}bb$ may be understood as the set of pairs of nodes (p, q) such that there are p_1 and p_2 with $p_1 \xrightarrow{a} p$, $p_1 \xrightarrow{b} p_2$, and $p_2 \xrightarrow{b} q$.

We extend the notion of inverse letters to inverse words by defining for every $u = x_1 \dots x_k \in \hat{\Sigma}^*$ the inverse \bar{u} of u by $\bar{u} = \bar{x}_k \dots \bar{x}_1$. Here, every x_i is an element of $\hat{\Sigma}$ and for $x_i = \bar{a}$, $a \in \Sigma$, \bar{x}_i is identified with a .

Given a word u over $\hat{\Sigma}$, we assign to u a normal form $u \downarrow$ which is obtained by removing all pairs $\bar{a}a$ or $a\bar{a}$ in u . Formally, we could define for Σ a rewrite system $\downarrow_{\Sigma} \subseteq \hat{\Sigma}^* \times \hat{\Sigma}^*$ by $\downarrow_{\Sigma} := \{(\bar{a}a, \varepsilon), (a\bar{a}, \varepsilon) \mid a \in \Sigma\}$ and show that it is terminating and confluent. Hence, we can speak also about *the* normal form of u .

Let us now define our first transformation. It is based on the notion of an extended substitution.

Definition 15.16. Let Σ and Γ be two alphabets. An **extended substitution** from Γ to Σ is a homomorphism from Γ into the power set of words over the extended alphabet $\hat{\Sigma}$. More precisely, h is a mapping such that for every $b \in \Gamma$

$$h(b) \in \mathcal{P}(\hat{\Sigma}^*)$$

and furthermore $h(\varepsilon) = \{\varepsilon\}$ and $h(uv) = h(u)h(v)$.

h is called **regular** iff $h(b)$ is a regular set for all $b \in \Gamma$, and **finite** iff $h(b)$ is a finite set for all $b \in \Gamma$.

Sometimes, we silently assume an extended substitution to be extended to a mapping from $\hat{\Gamma}^*$ to $\mathcal{P}(\hat{\Sigma}^*)$ by $h(\bar{b}) = \overline{h(b)}$ for $b \in \Gamma$.

Now we are ready to make precise the notion of an inverse substitution of a graph.

Definition 15.17. Let $\mathfrak{G} = (V, E)$ be a graph with edge labels from a given alphabet Σ . Furthermore, let Γ be an alphabet, and let $h : \Gamma \rightarrow \mathcal{P}(\hat{\Sigma}^*)$ be an extended substitution. We define the **inverse substitution** $h^{-1}(\mathfrak{G})$ to be the graph $\mathfrak{G}' = (V, E')$ such that

$$s \xrightarrow[\mathfrak{G}']{b} t \text{ iff } \exists u \in h(b) \ s \xrightarrow[\mathfrak{G}]{u} t$$

for all $s, t \in V$. The inverse substitution is called **regular** (respectively **finite**) iff h is a regular (respectively finite) extended finite substitution.

Example 15.18. Let $\Sigma = \{a\}$ be a singleton alphabet. Consider the extended substitution given by $h(a) = \{\bar{0}1\}$. The corresponding inverse substitution of the infinite binary tree is shown in Figure 15.4.

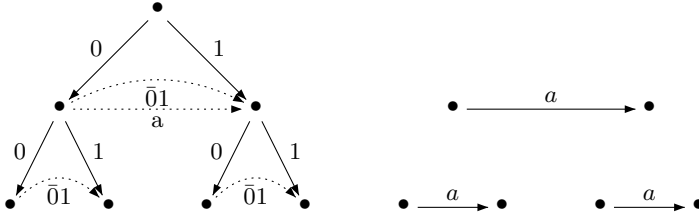


Fig. 15.4. A non-connected prefix-recognizable graph

For the graphs under consideration, we may assume without loss of generality that their nodes are words over some alphabet Σ . Hence, the nodes of our graphs constitute languages. A natural operation on languages is restriction. We will consequently also consider a second transformation on the binary tree called *restriction*.

Definition 15.19. Let $\mathfrak{G} = (V, E)$ be a graph with universe $V \subseteq N^*$ and edges $E \subseteq V \times \Sigma \times V$ for given alphabets N and Σ . Let L be a language over N . The **restriction** of \mathfrak{G} with respect to L is defined to be the graph

$$(V \cap L, E \cap (L \times \Sigma \times L))$$

and is denoted by $\mathfrak{G}|_L$. The restriction is called **regular** (respectively **finite**) iff L is a regular (respectively finite) set.

A subgraph of a given graph \mathfrak{G} can be identified by a restriction such that its nodes belong to the restricted language.

Let us show that regular restrictions of regular inverse substitutions are definable in MSO to obtain one of our main results:

Theorem 15.20. *Given a graph \mathfrak{G} with a unique root r , a regular substitution h , and a regular label language $L \in \text{REG}(\hat{N}^*)$, we have:*

$$\text{MTh}(\mathfrak{G}) \text{ decidable} \implies \text{MTh}(h^{-1}(\mathfrak{G})|_{L_{\mathfrak{G}}}) \text{ decidable}$$

where $L_{\mathfrak{G}} := \{s \mid r \xrightarrow[\mathfrak{G}]{L} s\}$.

Proof. Let φ be an MSO-formula. Observe that an a -successor of $h^{-1}(\mathfrak{G})|_{L_{\mathfrak{G}}}$ corresponds to an $h(a)$ -path in \mathfrak{G} . Furthermore, an element (a node) x exists in $h^{-1}(\mathfrak{G})|_{L_{\mathfrak{G}}}$ iff it is the starting point or end point of some path in \mathfrak{G} and is not removed because of the restriction with respect to L . The latter means that

the element is reached by some L -path from the root z of the graph. Hence, we define the formula $\varphi^{L,h,z}$ inductively:

$$\begin{aligned}
 S_a xy^{L,h,z} &= \text{Path}_{h(a)}(x, y) \\
 (x \in X)^{L,h,z} &= x \in X \\
 (\neg\varphi)^{L,h,z} &= \neg(\varphi^{L,h,z}) \\
 (\varphi \wedge \psi)^{L,h,z} &= \varphi^{L,h,z} \wedge \psi^{L,h,z} \\
 (\exists X\varphi)^{L,h,z} &= \exists X\varphi^{L,h,z} \\
 (\exists x\varphi)^{L,h,z} &= \exists x(\text{Path}_L(z, x) \wedge \exists y(\text{Path}_M(x, y) \vee \text{Path}_M(y, x)) \\
 &\quad \wedge \varphi^{L,h,z})
 \end{aligned}$$

where $M = \bigcup h(a)$ and $\text{Path}_L(x, y)$ is as in Lemma 15.13. It is easy to see that

$$h^{-1}(\mathfrak{G})|_{L_{\mathfrak{G}}} \models \varphi \text{ iff } \mathfrak{G} \models \exists z (\forall y \text{Path}_{N^*}(x, y) \wedge \varphi^{L,h,z}).$$

Note that the first conjunct assures that z is indeed a root of \mathfrak{G} .

The previous proof can easily be employed for defining an interpretation of a graph $h^{-1}(\mathfrak{G})|_{L_{\mathfrak{G}}}$ in \mathfrak{G} . The successor relations are explicitly given and the domain is easily defined using our ideas that led to the definition of the $\exists x$ case. The congruence $\varepsilon(x, y)$ can be defined to be equality. Thus, we see:

Corollary 15.21. *Regular restrictions of regular inverse substitutions of $\mathfrak{T}_{\mathbb{B}}$ yield prefix-recognizable graphs.*

We conclude that the graph shown in Example 15.18 is prefix-recognizable. Thus, we see that prefix-recognizable graphs are not necessarily connected. This distinguishes prefix-recognizable graphs from tree-like structures presented in Chapter 16.

As mentioned above, we will show in the next section that prefix-recognizable graphs are indeed the graphs obtained as regular restrictions of regular inverse substitutions of $\mathfrak{T}_{\mathbb{B}}$, establishing a further representation of the studied objects.

A further transformation considered is a *marking* of nodes belonging to a given set. To mark nodes of our graph, we introduce a new symbol $\# \notin \Sigma$, and, as we will see in the next section, add a $\#$ -edge for nodes to be marked. Therefore, we consider also paths including this symbol $\#$. To simplify our notation, we write $\hat{\Sigma}_{\#}$ for an extended alphabet together with the symbol $\#$. Also, we consider normalizations which further reduce $\#$ to the empty word ε . For a word $u \in \hat{\Sigma}_{\#}^*$ its corresponding normal form is denoted by $u \downarrow_{\#}$.

Definition 15.22. Let $\mathfrak{G} = (V, E)$ be a graph with universe $V \subseteq N^*$ and edges $E \subseteq V \times \Sigma \times V$ for given alphabets N and Σ . Let L be a language over N . The **marking** of \mathfrak{G} , with respect to L , by a new symbol $\#$ not in Σ is defined to be the graph

$$(V, E'), \text{ where } E' = E \cup \{s \xrightarrow{\#} s \mid s \in L\},$$

and is denoted by $\#_L(\mathfrak{G})$. The marking is called **regular** (respectively **finite**) iff L is a regular (respectively finite) set.

Instead of $\hat{\Sigma}$, we sometimes consider $\hat{\Sigma}_{\#}$. All definitions extend to this case in the obvious way.

Let us collect some properties and interrelations of the transformations mentioned above.

Lemma 15.23. *Let Σ, Γ , and Ξ be alphabets, and \mathfrak{G} be a Σ -graph. Let h be an extended substitution from Γ to Σ , and g one from Ξ to Γ . Then the following holds:*

- (1) $s \xrightarrow[h^{-1}(\mathfrak{G})]{u} t$ iff $s \xrightarrow[\mathfrak{G}]{h(u)} t$ for any $u \in \Gamma^+$ and $s, t \in V_{\mathfrak{G}}$.
- (2) $g^{-1}(h^{-1}(\mathfrak{G})) = ((g \circ h)^{-1}(\mathfrak{G}))|_{V_{h^{-1}(\mathfrak{G})}}$, and if $\varepsilon \notin g(\Xi)$ then $g^{-1}(h^{-1}(\mathfrak{G})) = ((g \circ h)^{-1}(\mathfrak{G}))$.

Definition 15.24. A set L is called **stable** in a graph \mathfrak{G} , iff any path between vertices in L contains only vertices in L :

$$\text{If } s_0 \xrightarrow[\mathfrak{G}]{} s_1 \cdots s_{n-1} \xrightarrow[\mathfrak{G}]{} s_n \text{ and } s_0, s_n \in L \text{ then } s_1, \dots, s_{n-1} \in L.$$

For example, L is stable in $\mathfrak{G}|_L$. A simple but useful insight is that a restriction to any stable set commutes with any inverse substitution.

Lemma 15.25. *If L is stable in \mathfrak{G} then*

$$h^{-1}(\mathfrak{G}|_L) = h^{-1}(\mathfrak{G})|_L.$$

Any restriction of an image of a graph is an image of a marking of the graph:

Lemma 15.26. *Let $\mathfrak{G} = (V, E)$ be a graph with universe $V \subseteq N^*$ and edges $E \subseteq V \times \Sigma \times V$ for given alphabets N and Σ . Let L be a language over N . Let $\#$ be a new symbol not in Σ . Furthermore, let h be an extended substitution from an alphabet Γ to Σ . Then*

$$(h^{-1}(\mathfrak{G}))|_L = g^{-1}(\#_L(\mathfrak{G})),$$

with $g(b) = \#h(b)\#$ for every $b \in \Gamma$.

Proof. By definition, $g^{-1}(\#_L(\mathfrak{G})) = \{s \xrightarrow[\#_L(\mathfrak{G})]{b} t \mid \exists u \in g(b).s \xrightarrow[\#_L(\mathfrak{G})]{u} t\}$. Since the words in the image under g are the ones under h with a $\#$ enclosing elements of L , we conclude that $\{s \xrightarrow[\#_L(\mathfrak{G})]{b} t \mid \exists u \in g(b).s \xrightarrow[\#_L(\mathfrak{G})]{u} t\} = \{s \xrightarrow[\#_L(\mathfrak{G})]{b} t \mid \exists u' \in h(b).s \xrightarrow[\#_L(\mathfrak{G})]{u'} t \text{ and } s, t \in L\}$. The latter is equal to $\{s \xrightarrow[\#_L(\mathfrak{G})]{b} t \mid \exists u' \in h(b).s \xrightarrow[\mathfrak{G}]{u'} t \text{ and } s, t \in L\}$ since the words in the images of h do not contain the symbol $\#$. Thus, we obtain $g^{-1}(\#_L(\mathfrak{G})) = h^{-1}(\mathfrak{G})|_L$.

We now show that the restriction to normal forms preserves regularity.

Lemma 15.27. *Let $L \in \text{REG}(N^*)$ and $M \in \text{REG}(\hat{N}_\#^*)$. Then we have in an effective way*

$$(L(\#_L(\mathfrak{T}_N)) \cap M) \downarrow_\# \in \text{REG}(N^*).$$

Proof. Let $M' = (L(\#_L(\mathfrak{T}_N)) \cap M) \downarrow_\#$. Let $\mathcal{A} = (Q, \hat{N}_\#, \delta, q_0, F)$ be a finite automaton recognizing M . We colour any vertex $u \in N^*$ of \mathfrak{T}_N by the set $c(u)$ of states p such that (p, u) is a vertex of the product $Q \times \#_L(\mathfrak{T}_N)$ reachable from (q_0, ε) :

$$c(u) = \{p \mid L((Q, \hat{N}_\#, \delta, q_0, \{p\})) \cap L(\#_L(\mathfrak{T}_N), \varepsilon, u) \neq \emptyset\}$$

Hence, $M' = \{u \in N^* \mid c(u) \cap F \neq \emptyset\}$. We show that M' is regular by proving that c is a regular colouring of $\#_L(\mathfrak{T}_N)$. We consider the following equivalence \equiv on N^* :

$$u \equiv v \text{ iff } c(u) = c(v) \text{ and } u^{-1}L = v^{-1}L$$

Note that $u^{-1}L$ is an abbreviation for $\{w \in N^* \mid uw \in L\}$. As the image of c is finite and L is regular, the equivalence \equiv is of finite index. Further, it is a simple matter to show that \equiv is right-regular. So $H := \{[u] \xrightarrow{a} [ua] \mid u \in N^* \text{ and } a \in N\}$ is finite and $M' = L(H, [\varepsilon], \{[u] \mid c(u) \cap F \neq \emptyset\})$ is regular. Here, $[u]$ denotes the equivalence class of u with respect to \equiv .

To show the effectiveness of the construction of M' , it suffices to show that H can be effectively constructed. The latter is clear if $c(u)$ is computable. This can be seen by recalling that

- (1) $L(\mathcal{A})$ is regular,
- (2) $L(\#_L(\mathfrak{T}_N), \varepsilon, u)$ is context-free for every $u \in N^*$,
- (3) and the intersection of a regular and a context-free language is context-free.

Hence, the emptiness of $L(\mathcal{A}) \cap L(\#_L(\mathfrak{T}_N), \varepsilon, u)$ is decidable.

Let us now show that instead of an arbitrary extended substitution, we can assume the image to be normalized in the following sense:

Proposition 15.28. *Let h be an extended substitution, yielding for each $a \in \Sigma$ a language over $\hat{N}_\#$, and $L \subseteq N^*$. Let \mathfrak{G} be the graph with all edges of the form $w(\overline{u \downarrow_\#}) \xrightarrow{a} w(v \downarrow_\#)$ for $wv \in h(a)$, $\overline{u}, v \in L(\#_{w^{-1}L}(\mathfrak{T}_N))$, and $w \in N^*$. Then*

$$h^{-1}(\#_L(\mathfrak{T}_N)) = \mathfrak{G}.$$

Proof. We show that $h^{-1}(\#_L(\mathfrak{T}_N)) \subseteq \mathfrak{G}$. Let $s \xrightarrow[h^{-1}(\#_L(\mathfrak{T}_N))]{a} t$. Thus, there is a $z \in h(a)$, such that $s \xrightarrow{z} t$. Let $w = s \sqcap t$, $w \in N^*$. Hence, $s \xrightarrow[\#_L(\mathfrak{T}_N)]{u} w \xrightarrow[\#_L(\mathfrak{T}_N)]{v} t$, with $wv = z$, and w is the node “closest to the root”. There are $x, y \in N^*$, such that $s = wx$ and $t = wy$, with $x \xrightarrow[\#_{w^{-1}L}(\mathfrak{T}_N)]{u} \varepsilon \xrightarrow[\#_{w^{-1}L}(\mathfrak{T}_N)]{v} y$. So, $\varepsilon \xrightarrow[\#_{w^{-1}L}(\mathfrak{T}_N)]{\overline{u}} x$,

$\bar{u} \in L(\#_{w^{-1}L}(\mathfrak{X}_N))$, and $x = \bar{u}\down_{\#} = \overline{u\down_{\#}}$. Similarly, $v \in L(\#_{w^{-1}L}(\mathfrak{X}_N))$ and $y = v\down_{\#}$. Thus, $s = wx = w\overline{u\down_{\#}}$ and $t = wy = w(v\down_{\#})$. Finally, we have $s \xrightarrow[\mathfrak{G}]{a} t$.

Let us now show the converse direction, i.e. $\mathfrak{G} \subseteq h^{-1}(\#_L(\mathfrak{X}_N))$. Consider $s \xrightarrow[\mathfrak{G}]{a} t$. There are $uv \in h(a)$ and $w \in N^*$, such that $\bar{u}, v \in L(\#_{w^{-1}L}(\mathfrak{X}_N))$, $s = w\overline{u\down_{\#}}$, $t = w(v\down_{\#})$. We must show that $s \xrightarrow[h^{-1}(\#_L(\mathfrak{X}_N))]{a} t$. Since $\bar{u} \in L(\#_{w^{-1}L}(\mathfrak{X}_N))$, we have $\varepsilon \xrightarrow[\#_{w^{-1}L}(\mathfrak{X}_N)]{\bar{u}} \bar{u}\down_{\#} = \overline{u\down_{\#}}$. So, $\overline{u\down_{\#}} \xrightarrow[\#_{w^{-1}L}(\mathfrak{X}_N)]{u} \varepsilon$, $s = w\overline{u\down_{\#}} \xrightarrow[\#_L(\mathfrak{X}_N)]{u} w$. In a similar manner, we show that $w \xrightarrow[\#_L(\mathfrak{X}_N)]{v} w(v\down_{\#}) = t$. Thus, $s \xrightarrow[\#_L(\mathfrak{X}_N)]{uv} t$, $s \xrightarrow[\#_L(\mathfrak{X}_N)]{h(a)} t$, and $s \xrightarrow[h^{-1}(\#_L(\mathfrak{X}_N))]{a} t$.

Proposition 15.28 is rather technical, but allows a simple presentation if we introduce further notation.

Definition 15.29. Let Σ and N be alphabets, $L \subseteq N^*$, and \mathfrak{G} a graph, whose edge relation is a subset of $N^* \times \Sigma \times N^*$. The **right concatenation** of \mathfrak{G} by L is the graph

$$\mathfrak{G}.L := \{uw \xrightarrow{a} vw \mid u \xrightarrow[\mathfrak{G}]{a} v \text{ and } w \in L\}$$

Similarly, we define their **left concatenation** $L.\mathfrak{G}$. For the sake of brevity, we also write $\mathfrak{G}L$ and $L\mathfrak{G}$ instead of $\mathfrak{G}.L$ and $L.\mathfrak{G}$, respectively.

It is folklore that a regular language $L \in \text{REG}(N^*)$ is the union of finitely many equivalence classes, given in the following form:

$$[u]_L := \{v \mid v^{-1}L = u^{-1}L\} \text{ and } [L] := \{[u]_L \mid u \in N^*\}$$

Now, Proposition 15.28 can be stated for regular extended substitutions in the following way:

Corollary 15.30. For any substitution $h : \Sigma \rightarrow \hat{N}_{\#}^*$ and $L \in \text{REG}(N^*)$, we have that $h^{-1}(\#_L(\mathfrak{X}_N))$ equals

$$\bigcup_{W \in [L]} W\{\overline{u\down_{\#}} \xrightarrow{a} v\down_{\#} \mid uv \in h(a) \text{ and } \bar{u}, v \in L(\#_{W^{-1}L}(\mathfrak{X}_N))\}.$$

If we further omit markings, we can simplify Proposition 15.28 to:

Corollary 15.31. For any substitution $h : \Sigma \rightarrow \hat{N}^*$ we have that

$$h^{-1}(\mathfrak{X}_N) = N^*\{u \xrightarrow{a} v \mid \bar{u}v \in h(a)\down, u, v \in N^* \text{ and } a \in \Sigma\}.$$

15.5 Representations of Prefix-Recognizable Graphs

In Section 15.4, we introduced inverse substitution, restriction, and marking as transformations on graphs. Here, we introduce classes of graphs as regular restrictions of regular inverse substitutions on the complete binary tree. Furthermore, we show that these classes can be obtained by considering regular inverse substitutions of regular markings on the binary tree. Additionally, we prove that any complete tree can be employed instead of the binary one. Last but not least, we give a representation in terms of prefix-transition graphs which provides a link to Section 15.3.

Definition 15.32. Let Σ and N be alphabets. We define the classes $\text{PRG}_N(\Sigma)_\downarrow$ and $\text{PRG}_N^\#(\Sigma)$ of graphs with edge labels over Σ and nodes in N^* :

- $\mathfrak{G} \in \text{PRG}_N(\Sigma)_\downarrow$ iff \mathfrak{G} is isomorphic to $h^{-1}(\mathfrak{X}_N)_\downarrow L$ for a suitable regular extended substitution h from Σ to N and $L \in \text{REG}(N^*)$.
- $\mathfrak{G} \in \text{PRG}_N^\#(\Sigma)$ iff \mathfrak{G} is isomorphic to $h^{-1}(\#_L(\mathfrak{X}_N))$ for a suitable regular extended substitution h from Σ to N and $L \in \text{REG}(N^*)$.

Proposition 15.33. For every alphabet N with $\mathbb{B} \subseteq N$, we have

$$\text{PRG}_N(\Sigma)_\downarrow = \text{PRG}_\mathbb{B}(\Sigma)_\downarrow = \text{PRG}_\mathbb{B}^\#(\Sigma) = \text{PRG}_N^\#(\Sigma)$$

Proof. We show that $\text{PRG}_N(\Sigma)_\downarrow \stackrel{(1)}{\subseteq} \text{PRG}_N^\#(\Sigma) \stackrel{(2)}{\subseteq} \text{PRG}_\mathbb{B}(\Sigma)_\downarrow \stackrel{(3)}{\subseteq} \text{PRG}_N(\Sigma)_\downarrow$. This shows $\text{PRG}_N(\Sigma)_\downarrow = \text{PRG}_\mathbb{B}(\Sigma)_\downarrow = \text{PRG}_\mathbb{B}^\#(\Sigma)$. The last equation implies for $N = \mathbb{B}$ also $\text{PRG}_\mathbb{B}(\Sigma)_\downarrow = \text{PRG}_\mathbb{B}^\#(\Sigma)$.

- (1) First we show $\text{PRG}_N(\Sigma)_\downarrow \subseteq \text{PRG}_N^\#(\Sigma)$. Let $\mathfrak{G} \in \text{PRG}_N(\Sigma)_\downarrow$. So \mathfrak{G} is isomorphic to $h^{-1}(\mathfrak{X}_N)_\downarrow L$ for a regular extended substitution h from Σ to N and $L \in \text{REG}(N^*)$. By Lemma 15.26, $h^{-1}(\mathfrak{X}_N)_\downarrow L = g^{-1}(\#_L(\mathfrak{X}_N))$ with $g(a) = \#h(a)\#$ for all $a \in \Sigma$.
- (2) We now show $\text{PRG}_N^\#(\Sigma) \subseteq \text{PRG}_\mathbb{B}(\Sigma)_\downarrow$. Let $\mathfrak{G} \in \text{PRG}_N^\#(\Sigma)$. So \mathfrak{G} is isomorphic to $h^{-1}(\#_L(\mathfrak{X}_N))$ for a regular extended substitution h from Σ to N and $L \in \text{REG}(N^*)$.

Let $\mathcal{A}_L = (Q, N, \delta, q_0, F)$ be a finite and complete deterministic automaton recognizing L . Without loss of generality, we may assume that for $p, q \in Q$ we have that $\delta(p, a) = \delta(q, b)$ implies $a = b$.² Thus, every reachable state of Q has a unique “incoming” letter a . Let P denote the set of all sequences of states which can be obtained from the initial state q_0 via δ : $P = \{q_0 p_1 \dots p_k \mid \exists k \geq 0, \exists a_1, \dots, a_k \in N, \delta(q_0, a_1) = p_1, \delta(p_{i-1}, a_i) = p_i \text{ for all } i \in \{1, \dots, k\}\}$. So every such sequence $q_0 p_1 \dots p_k$ corresponds to a unique word $a_1 \dots a_k$. Furthermore, it is easy to see that P is a regular set. Now consider the finite and therefore regular extended substitution f defined by

$$\begin{aligned} f(a) &= \{\bar{p}p q \mid \delta(p, a) = q\} \\ f(\#) &= \{\bar{p}p \mid p \in F\} \end{aligned}$$

² Otherwise, duplicate states of \mathcal{A}_L appropriately.

Then: $\#_L(\mathfrak{T}_N)$ is isomorphic to $f^{-1}(\mathfrak{T}_Q)|_P$. Instead of giving a formal proof, let us consider as an example the transition $s \xrightarrow{\#} s$. Then $s = a_1 \dots a_k \in L$. There is a unique sequence $q_0 p_1 \dots p_k$ with $\delta(q_0, a_1) = p_1$, $\delta(p_{i-1}, a_i) = p_i$ for $i \in \{1, \dots, k\}$, $p_k \in F$, and $q_0 p_1 \dots p_k \in P$. Hence, there is a corresponding path from node q_0 to node $q_0 p_1 \dots p_k$ in $f^{-1}(\mathfrak{T}_Q)|_P$ labelled by $a_1 \dots a_k$. This situation is depicted in Figure 15.5.

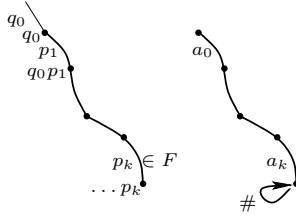


Fig. 15.5. Words vs. State

By definition, P is the vertex set of the connected component of $f^{-1}(\mathfrak{T}_Q)$ containing q_0 . Hence, P is stable for $f^{-1}(\mathfrak{T}_Q)$. We could now easily prove that \mathfrak{G} (isomorphic to $h^{-1}(\#_L(\mathfrak{T}_N))$) is isomorphic to $h^{-1}(f^{-1}(\mathfrak{T}_Q))|_P$. However, we want to achieve such a result for \mathbb{B} instead of Q . Therefore, we use a standard encoding of elements of Q by sequences of zeros and ones. Let $Q = \{p_1, \dots, p_n\}$ and for $i \in \{1, \dots, n\}$

$$g(p_i) = 01^{i-1}.$$

Furthermore, let $M = g(\{p_1, \dots, p_n\}^*) = \{0, \dots, 01^{n-1}\}^*$. Note that M is stable for $g^{-1}(\mathfrak{T}_{\mathbb{B}})$. Then we have

$$g[\mathfrak{T}_Q] = g^{-1}(\mathfrak{T}_{\mathbb{B}})|_M.$$

The latter means that \mathfrak{T}_Q is isomorphic to $g^{-1}(\mathfrak{T}_{\mathbb{B}})|_M$ (via the isomorphism g). Figure 15.6 depicts an encoding of the ternary tree in the binary tree. The corresponding encoding function is given by:

$$\begin{aligned} g(a_0) &= 0 \\ g(a_1) &= 01 \\ g(a_2) &= 011 \end{aligned}$$

Since $\#_L(\mathfrak{T}_N)$ is isomorphic to $f^{-1}(\mathfrak{T}_Q)|_P$, it is also isomorphic to the isomorphic image of $f^{-1}(\mathfrak{T}_Q)|_P$ via g . Hence, it is isomorphic to:

$$\begin{aligned} g[f^{-1}(\mathfrak{T}_Q)|_P] &= g[f^{-1}(\mathfrak{T}_Q)]|_{g(P)} \\ &= f^{-1}(g[\mathfrak{T}_Q])|_{g(P)} \\ &= f^{-1}(g^{-1}(\mathfrak{T}_{\mathbb{B}})|_M)|_{g(P)} \\ &= f^{-1}(g^{-1}(\mathfrak{T}_{\mathbb{B}}))|_{M \cap g(P)} && \text{Lemma 15.25} \\ &= (f \circ g)^{-1}(\mathfrak{T}_{\mathbb{B}})|_{V_{g^{-1}(\mathfrak{T}_{\mathbb{B}})} \cap M \cap g(P)} && \text{Lemma 15.23} \\ &= (f \circ g)^{-1}(\mathfrak{T}_{\mathbb{B}})|_{g(P)} \end{aligned}$$

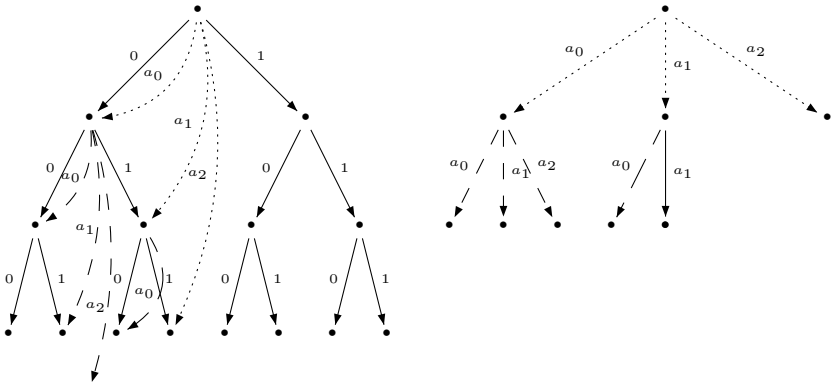


Fig. 15.6. Encoding a ternary tree in the binary tree

Note that $g(P)$ is stable for $(f \circ g)^{-1}(\mathfrak{T}_{\mathbb{B}})$, and, using Lemma 15.25, \mathfrak{G} is isomorphic to:

$$\begin{aligned} & h^{-1}((f \circ g)^{-1}(\mathfrak{T}_{\mathbb{B}})|_{g(P)}) \\ &= (h \circ f \circ g)^{-1}(\mathfrak{T}_{\mathbb{B}})|_{g(P)} \\ &= ((h \circ f \circ g)\downarrow_{\#})^{-1}(\mathfrak{T}_{\mathbb{B}})|_{g(P)} \end{aligned}$$

where for any $x \in \Sigma$, $((h \circ f \circ g)\downarrow_{\#})(x) = ((f \circ g)(h(x)))\downarrow_{\#}$. As $f \circ g$ is a finite substitution, we have $((h \circ f \circ g)\downarrow_{\#})$ is a regular extended substitution. Hence, $\mathfrak{G} \in \text{PRG}_{\mathbb{B}}(\Sigma)_|$.

- (3) Finally, we show $\text{PRG}_{\mathbb{B}}(\Sigma)_| \subseteq \text{PRG}_N(\Sigma)_|$. Let $\mathfrak{G} \in \text{PRG}_{\mathbb{B}}(\Sigma)_|$. So \mathfrak{G} is isomorphic to $h^{-1}(\mathfrak{T}_{\mathbb{B}})|_L$ for a regular extended substitution $h : \Sigma \rightarrow \mathbb{B}$ and $L \in \text{REG}(\mathbb{B}^*)$. We have $\mathfrak{T}_{\mathbb{B}} = \iota^{-1}(\mathfrak{T}_N)|_{\mathbb{B}^*}$, where ι denotes the identity on \mathbb{B} . Note that \mathbb{B}^* is stable for $\iota^{-1}(\mathfrak{T}_N)$. Hence, \mathfrak{G} is isomorphic to:

$$\begin{aligned} h^{-1}(\iota^{-1}(\mathfrak{T}_N)|_{\mathbb{B}^*})|_L &= (h^{-1}(\iota^{-1}(\mathfrak{T}_N))|_{\mathbb{B}^*})|_L && \text{by Lemma 15.25} \\ &= (h \circ \iota)^{-1}(\mathfrak{T}_N)|_{V_{\iota^{-1}(\mathfrak{T}_N)} \cap \mathbb{B}^* \cap L} && \text{by Lemma 15.23} \\ &= (h \circ \iota)^{-1}(\mathfrak{T}_N)|_{\mathbb{B}^* \cap L} \end{aligned}$$

Since $h \circ \iota$ is a regular extended substitution, we have $\mathfrak{G} \in \text{PRG}_{\mathbb{B}}(\Sigma)_|$.

We now give three important representations of prefix-recognizable graphs.

Theorem 15.34 ([28]). *Given an alphabet N with at least two letters, the following properties are equivalent:*

- (1) \mathfrak{G} is interpretable in $\mathfrak{T}_{\mathbb{B}}$.
- (2) $\mathfrak{G} \in \text{PRG}_N(\Sigma)_|$.
- (3) \mathfrak{G} is isomorphic to $(N^*H)|_L$ for some recognizable $H \subseteq N^* \times \Sigma \times N^*$ and $L \in \text{REG}(N^*)$.
- (4) \mathfrak{G} is isomorphic to $\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$ for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(N^*)$.

Proof. (2) \Rightarrow (3): Assume $\mathfrak{G} \in \text{PRG}_N(\Sigma)_|$. So \mathfrak{G} is isomorphic to $h^{-1}(\mathfrak{T}_N)_{|L}$ for an appropriate extended substitution h from Σ to N and $L \in \text{REG}(N^*)$. By Corollary 15.31, we can write $h^{-1}(\mathfrak{T}_N)$ as N^*H for

$$H = \{\bar{u}v \in h(a) \downarrow \cap \overline{N^*}N^*, a \in \Sigma\}.$$

Since h is regular, $h(a)$ is a regular language, let us say C . We are done by showing that $C \downarrow \cap \overline{N^*}N^*$ is a finite union of the form UV for $U, V \in \text{REG}(N^*)$. Let $\mathcal{A} = (Q, \hat{N}, \delta, q_0, F)$ be the automaton recognizing C . It is easy to see that $C \downarrow \cap \overline{N^*}N^*$ equals

$$\bigcup_{q \in Q} \overline{(L(Q, N, \delta, q_0, q) \cap L(\mathfrak{T}_N)) \downarrow} \overline{(L(Q, N, \delta, q, F) \cap L(\mathfrak{T}_N)) \downarrow},$$

and regularity follows from Lemma 15.27.

(2) \Rightarrow (4): Consider $\mathfrak{G} \in \text{PRG}_N(\Sigma)_|$. Hence, \mathfrak{G} is isomorphic to a $h^{-1}(\mathfrak{T}_N)_{|L}$ for a suitable regular extended substitution from Σ to N and $L \in \text{REG}(N^*)$. For every $a \in \Sigma$, let $\mathcal{A}_a = (Q_a, \hat{N}_\#, \delta_a, q_{0a}, F_a)$ be the automaton recognizing $h(a)$. By Corollary 15.30 and similar arguments as in the previous case, we can write $h^{-1}(\mathfrak{T}_N)$ as

$$\bigcup_{\substack{W \in [L] \\ a \in \Sigma \\ q \in Q_a}} W(U(a, q) \xrightarrow{a} V(a, q)),$$

where

$$U(a, q) = \overline{(L(Q_a, \hat{N}_\#, \delta_a, q_{0a}, q) \cap L(\#_{W^{-1}L}(\mathfrak{T}_N))) \downarrow}_\#$$

and

$$V(a, q) = \overline{(L(Q_a, \hat{N}_\#, \delta_a, q, F_a) \cap L(\#_{W^{-1}L}(\mathfrak{T}_N))) \downarrow}_\#.$$

Regularity again follows from Lemma 15.27.

(3) \Rightarrow (2): Let $H \subseteq N^* \times \Sigma \times N^*$ be a recognizable graph and $L \in \text{REG}(N^*)$. By Corollary 15.31, $H = h^{-1}(\mathfrak{T}_N)$, such that $h(a) = \{\bar{u}v \mid u \xrightarrow{a} v\}$ for every $a \in \Sigma$.

(4) \Rightarrow (2): Let \mathfrak{G} be isomorphic to $\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$ for some $n \geq 1$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(N^*)$. Define L to be the regular language $L = \bigcup_{i=1}^n W_i a_i$, and let $h(a) = \bigcup \{\overline{U_i a_i \# \bar{a}_i V_i} \mid a_i = a\}$ for every $a \in \Sigma$ be a regular extended substitution. Then $\mathfrak{G} = h^{-1}(\#_L(\mathfrak{T}_N))$.

We have shown that (2) – (4) are equivalent. We had already shown in Theorem 15.14 that (1) implies (4) and (2) implies (1), using Proposition 15.33. Thus, all equivalences are shown.

15.6 Automata for Prefix-Recognizable Graphs

Let us conclude this chapter with a simple link from prefix-recognizable graphs to pushdown automata. The result is due to Stirling [168], and the proof is due to Caucal.³

Theorem 15.35. *Regular restrictions of the ε -closures of pushdown graphs are prefix-recognizable graphs.*

Proof. “ \Rightarrow ”: Let us consider ε to be a new symbol. Then, the transition graph of a pushdown automaton with ε -transitions is a regular graph. The ε -closure of this graph can be obtained by an inverse regular mapping. Since regular graphs are a special kind of prefix-recognizable graphs and the class of prefix-recognizable graphs is closed with respect to regular inverse substitutions, we have that the latter is prefix-recognizable.

“ \Leftarrow ”: Let $\mathfrak{G} = (\bigcup_{i=1}^n (U_i \xrightarrow{a_i} V_i) N^*)|_L$ be a prefix-recognizable graph with $U_1, V_1, \dots, U_n, V_n, L \in \text{REG}(N^*)$. For each U_i and V_i we have finite automata $\mathcal{A}_i^U = (Q_i^U, N, \delta_i^U, q_{0i}^U, F_i^U)$ and $\mathcal{B}_i^V = (Q_i^V, N, \delta_i^V, q_{0i}^V, F_i^V)$ recognizing U_i and V_i , respectively. Assume that these automata have pairwise disjoint state sets. Let $\#$ be a new symbol and construct the following rewriting system R :

$$\begin{aligned} \# &\xrightarrow{\varepsilon} q_{0i}^U \\ pa &\xrightarrow{\varepsilon} q \quad \text{if } q \in \delta_i^U(p, a) \\ p &\xrightarrow{\varepsilon} q \quad \text{if } p \in F_i^U \text{ and } q \in F_j^U \\ q &\xrightarrow{\varepsilon} pa \quad \text{if } q \in \delta_i^V(p, a) \\ q_{0i}^V &\xrightarrow{a_i} \# \end{aligned}$$

So \mathfrak{G} is equal to the restriction to $\#_L$ of the ε -closure of the prefix transition graph of R .

15.7 Conclusion

In this chapter we introduced the class of prefix-recognizable graphs, originally introduced by Caucal (cf. [28]). We have shown that this class of graphs is the largest class of graphs providing a decidable MSO-theory provable by interpretation in the infinite binary tree.

Several further representations of prefix-recognizable graphs were given in the literature. Let us sum up (some) known results in the following theorem. Whenever the formal notions are not clear, we refer the reader to the citations given.

Theorem 15.36. *Let \mathfrak{G} be a graph. The following statements are equivalent:*

- (1) $\mathfrak{G} = h^{-1}(\mathfrak{T}_{\mathbb{B}})|_C$ for a regular substitution h and a regular language C .
- (2) \mathfrak{G} is isomorphic to $\bigcup_{i=1}^n W_i(U_i \xrightarrow{a_i} V_i)$ for some $n \geq 0$; $a_1, \dots, a_n \in \Sigma$; $U_1, V_1, W_1, \dots, U_n, V_n, W_n \in \text{REG}(N^*)$.

³ private communication

- (3) $\mathfrak{G} = h^{-1}(\#_C(\mathfrak{T}_{\mathbb{B}}))$ for a regular substitution h and a regular marking C .
- (4) \mathfrak{G} is MSO-interpretable in the binary tree $\mathfrak{T}_{\mathbb{B}}$.
- (5) \mathfrak{G} is VR-equational.
- (6) \mathfrak{G} is a prefix-transition graph of Type-2.
- (7) \mathfrak{G} is the configuration graph of a pushdown automaton with ε -transitions.

The equivalence of (1) – (3) was obtained by Caucal in [28]. (4) and (5) are shown in [6] and [12]. The last two characterizations are due to Stirling [168]. In this chapter, we have shown the equivalence of (1) – (4) and (7).

Two-player games for push-down graphs and prefix-recognizable are studied Chapter 17. A different natural class of objects providing a decidable MSO-theory is presented in Chapter 16.