

A NOTE ON CONTEXT-FREE LANGUAGES

R.F.C. WALTERS*

Pure Mathematics Department, University of Sydney, N.S.W. 2006, Australia

Communicated by G.M. Kelly

Received 4 November 1988

This paper describes regular and context-free grammars as certain morphisms of graphs, and the associated languages in terms of appropriate free category constructions applied to these graphs.

Introduction

This note contains a description of regular languages in terms of the notion of free category on a reflexive graph, and of context-free languages in terms of the notion of free category with products on a multigraph. More precisely, in each case a grammar is a morphism $\phi : \mathbf{G} \rightarrow \mathbf{H}$ of the appropriate kind of graph. Then arrows in the appropriate free category $\mathcal{F}\mathbf{G}$ are prescriptions for the construction of strings, while arrows on the appropriate free category $\mathcal{F}\mathbf{H}$ are simply strings in the language. The language defined by the grammar is the set of strings in the image of $\mathcal{F}\phi$. The problem of parsing is the problem of finding the inverse image of a string under $\mathcal{F}\phi$.

An early reference for context-free languages is [1]; a reference for category theory, which gives an exposition of Lawvere's work on the relation between calculi of terms and categories with products, is [3]; an analysis of paragraphs in terms of morphisms of reflexive graphs, analogous to my description below of regular grammars, appears in [2].

1. Regular languages

A *reflexive graph* (or 1-dimensional simplicial set) is a pair of sets $\mathbf{G}_0, \mathbf{G}_1$ and three functions

$$d_0, d_1 : \mathbf{G}_1 \rightarrow \mathbf{G}_0, \quad s : \mathbf{G}_0 \rightarrow \mathbf{G}_1$$

such that $d_0s = 1_{\mathbf{G}_0}$, $d_1s = 1_{\mathbf{G}_0}$. The elements of \mathbf{G}_0 are called vertices or *objects*; the

* The author gratefully acknowledges the support of the Australian Research Council.

elements of \mathbf{G}_1 are called directed edges or *arrows*. If $a \in \mathbf{G}_1$ and $d_0 a = X$, $d_1 a = Y$ then we write $\alpha : X \rightarrow Y$ and call X the *domain* of α , Y the *codomain* of α ; we denote sX by 1_X (the *identity* arrow of X). A *morphism* $\phi : \mathbf{G} \rightarrow \mathbf{H}$ of *reflexive graphs* is a pair of functions $\phi_0 : \mathbf{G}_0 \rightarrow \mathbf{H}_0$, $\phi_1 : \mathbf{G}_1 \rightarrow \mathbf{H}_1$ which preserves the domain, co-domain and identity.

There is an obvious forgetful functor \mathcal{U} from the category \mathcal{CAT} of categories to the category $\mathcal{GRPH}_{\text{ref}}$ of reflexive graphs, with a left adjoint \mathcal{F} . The objects of \mathcal{FG} are the same as those of \mathbf{G} ; the arrows from X to Y in \mathcal{FG} are either identity arrows (if $X = Y$) or directed non-empty paths of non-identity arrows of \mathbf{G} , beginning at X and ending at Y .

Given an alphabet \mathbf{A} there is an associated reflexive graph $\tilde{\mathbf{A}}$ with only one object I , and with arrows from I to I being the elements of \mathbf{A} together with the identity arrow ε . Notice that a morphism $\mathbf{G} \rightarrow \tilde{\mathbf{A}}$ just means a labelling of each arrow of \mathbf{G} by an element of \mathbf{A} or by the identity arrow ε . Further, $\mathcal{F}\tilde{\mathbf{A}}$ has one object I , and $\text{Hom}_{\mathcal{F}\tilde{\mathbf{A}}}(I, I)$ is the free monoid \mathbf{A}^* on \mathbf{A} .

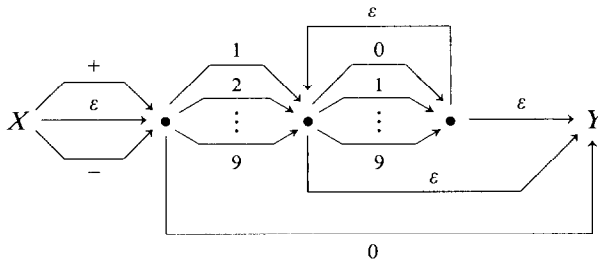
Definition. A *regular grammar* on a finite alphabet \mathbf{A} is a morphism of finite reflexive graphs

$$\phi : \mathbf{G} \rightarrow \tilde{\mathbf{A}}.$$

Given two object X, Y in \mathbf{G} there is an associated subset of \mathbf{A}^* , namely $\mathcal{F}\phi(\text{Hom}_{\mathcal{F}\mathbf{G}}(X, Y))$. Subsets obtained in this way from regular grammars are called *regular languages*.

The meaning of this definition, and its relation with standard notions, is best clarified by the examination of an illustrative example.

Example. Let \mathbf{A} be the set of digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 together with the signs +, -. Then the following labelled diagram represents a morphism from a reflexive graph \mathbf{G} to $\tilde{\mathbf{A}}$:



The graph \mathbf{G} is what remains when the labels are removed, and identity arrows are added (the identity arrows of \mathbf{G} have been suppressed in the diagram). The reason that reflexive graphs are considered is that while only the non-identity arrows

of \mathbf{G} are displayed in the diagram, some of these are labelled with the identity arrow of $\bar{\mathbf{A}}$.

Now some typical paths from X to Y in $\mathcal{F}\mathbf{G}$ (writing composition from left to right) are

$$+73\varepsilon8\varepsilon9\varepsilon0\varepsilon \quad \text{and} \quad \varepsilon73\varepsilon8\varepsilon9\varepsilon0\varepsilon \quad \text{and} \quad -0$$

and their images under $\mathcal{F}\phi$ are

$$+73890 \quad \text{and} \quad 73890 \quad \text{and} \quad -0$$

respectively. It is clear that the subset of \mathbf{A}^* defined by this grammar consists of integers (or arbitrary length) with leading zeros suppressed and with an optional plus or minus sign.

2. Context-free languages

A *multigraph* \mathbf{G} is a sequence of sets $\mathbf{G}_*, \mathbf{G}_0, \mathbf{G}_1, \mathbf{G}_2, \dots$ and for $n = 0, 1, 2, \dots$ functions

$$d_1, d_2, \dots, d_n, c: \mathbf{G}_n \rightarrow \mathbf{G}_*.$$

The elements of \mathbf{G}_* are called vertices or *objects*; the elements of \mathbf{G}_n are called directed edges or *arrows*. If α is in \mathbf{G}_n and $d_i\alpha = X_i$, $c\alpha = Y$, then we write $\alpha: X_1X_2\cdots X_n \rightarrow Y$ and call $X_1X_2\cdots X_n$ the *domain* of α , Y the *codomain* of α ; when α is in \mathbf{G}_0 and $c\alpha = Y$ we write $\alpha: 1 \rightarrow Y$. A *morphism* $\phi: \mathbf{G} \rightarrow \mathbf{H}$ of *multigraphs* is a sequence of functions $\phi_*: \mathbf{G}_* \rightarrow \mathbf{H}_*$, $\phi_0: \mathbf{G}_0 \rightarrow \mathbf{H}_0$, $\phi_1: \mathbf{G}_1 \rightarrow \mathbf{H}_1, \dots$ which preserves the operations d_1, d_2, \dots and c .

There is an forgetful functor \mathcal{U}_\times from the category \mathcal{CAT}_\times of categories with assigned strictly-associative finite products (and functors preserving the assigned products) to the category \mathcal{MULT} of multigraphs. If \mathbf{C} is such a category with products, then the objects of $\mathcal{U}_\times\mathbf{C}$ are the objects of \mathbf{C} , and the arrows of $\mathcal{U}_\times\mathbf{C}_n$ are the arrows in \mathbf{C} from an assigned n -ary product $X_1 \times X_2 \times X_3 \times \cdots \times X_n$ of objects of \mathbf{C} to a single object Y of \mathbf{C} . The functor \mathcal{U}_\times has a left adjoint \mathcal{F}_\times . The objects of $\mathcal{F}_\times\mathbf{G}$ are strings of objects in \mathbf{G} ; the arrows of $\mathcal{F}_\times\mathbf{G}$ from $X_1X_2X_3\cdots X_n$ to $Y_1Y_2\cdots Y_m$ are m -tuples of *terms* and composition is substitution of terms. To be more explicit, arrows of $\mathcal{F}_\times\mathbf{G}$ are defined inductively as follows. For each object X of \mathbf{G} take an infinite sequence x_1, x_2, x_3, \dots of *variables* of that type. Then

(i) x_i is an arrow in $\mathcal{F}_\times\mathbf{G}$ from any string containing at least i occurrences of X to X ,

(ii) given for each $j = 1, 2, 3, \dots, m$ an arrow $\alpha_j: S \rightarrow X_j$ in $\mathcal{F}_\times\mathbf{G}$, where S is a string and X_j is an object of \mathbf{G} , then $\alpha_1, \alpha_2, \dots, \alpha_m$ is an arrow in $\mathcal{F}_\times\mathbf{G}$ from S to $X_1X_2\cdots X_m$,

(iii) given $\alpha: S \rightarrow X_1X_2\cdots X_n$ an arrow in $\mathcal{F}_\times\mathbf{G}$ and $\beta: X_1X_2\cdots X_n \rightarrow Y$ an arrow in \mathbf{G} , then $\beta(\alpha)$ is an arrow in $\mathcal{F}_\times\mathbf{G}$ from S to Y .

Given an alphabet \mathbf{A} there is an associated multigraph $\bar{\mathbf{A}}$ with only one object M ,

and with one arrow μ_n from M^n to M for each n , together with the elements of \mathbf{A} as arrows from 1 to M . Notice that a morphism $\mathbf{G} \rightarrow \tilde{\mathbf{A}}$ just means a labelling of each arrow of \mathbf{G} by an element of \mathbf{A} (for arrows in \mathbf{G}_0) or by the arrow $\mu_n: M^n \rightarrow M$.

Further, $\mathcal{F}_\times \tilde{\mathbf{A}}$ has objects M^n ($n=0, 1, 2, 3, \dots$). To see what the arrows in $\mathcal{F}_\times \tilde{\mathbf{A}}$ are like, consider the following three arrows from M^3 to M :

$$\mu_2(\mu_2(x_1, x_2), x_3) \quad \text{and} \quad \mu_3(x_1, x_2, x_3) \quad \text{and} \quad \mu_2(x_1, \mu_2(x_2, x_3))$$

where x_1, x_2, x_3 are variables of type M . These arrows may be identified with the three different bracketings of x_1, x_2, x_3 . More generally, arrows in $\mathcal{F}_\times \tilde{\mathbf{A}}$ may be identified with bracketings of variables and elements of the alphabet \mathbf{A} .

Consider $\mathbf{Mon}_\mathbf{A}$, the algebraic theory of monoids augmented by the set \mathbf{A} of constants. $\mathbf{Mon}_\mathbf{A}$ is a quotient category of $\mathcal{F}_\times \tilde{\mathbf{A}}$; the quotient (product preserving) functor ψ from $\mathcal{F}_\times \tilde{\mathbf{A}}$ to $\mathbf{Mon}_\mathbf{A}$ is the identity on objects, and identifies different bracketings which are equivalent under associativity. It is clear that $\text{Hom}_{\mathbf{Mon}_\mathbf{A}}(1, M)$ is the free monoid \mathbf{A}^* on \mathbf{A} .

Definition. A *context-free grammar*, on a finite alphabet \mathbf{A} is a morphism of multigraphs

$$\phi: \mathbf{G} \rightarrow \tilde{\mathbf{A}},$$

where \mathbf{G} is a multigraph with only a finite number of objects and arrows. Given an object E in \mathbf{G} there is an associated subset of \mathbf{A}^* , namely

$$\psi \mathcal{F}_\times \phi(\text{Hom}_{\mathcal{F}_\times \mathbf{G}}(1, E)).$$

Subsets obtained in this way from context-free grammars are called *context-free languages*.

Again the meaning of the definition, and its relation with standard notions, is best clarified by the examination of an illustrative example.

Example. Let \mathbf{A} be the set of characters a, b, c, \dots, x, y, z together with the symbols $+, [,]$. Let \mathbf{G}_* be the set E, L, R, S, C . Then the following diagram represents a morphism from a multigraph \mathbf{G} to $\tilde{\mathbf{A}}$:

$$\begin{array}{ll} a, b, c, \dots, z: 1 \rightarrow C, & \mu_3: ESE \rightarrow E, \\ [: 1 \rightarrow L, & \mu_3: LER \rightarrow E, \\]: 1 \rightarrow R, & \mu_1: C \rightarrow E. \\ +: 1 \rightarrow S, \end{array}$$

The names given to the arrows in the diagram are the labels (in this example no ambiguity arises from naming the arrows by the labels).

Now a typical arrow from 1 to E in $\mathcal{F}_\times \mathbf{G}$ is

$$\mu_3(\mu_3([\mu_3(\mu_1(a), +, \mu_1(b)), 1], +, \mu_1(a)).$$

The image under $\psi\mathcal{F}\phi$ is

$$[a + b] + a.$$

It is clear that the subset of \mathbf{A}^* defined by this grammar consists of arithmetic expressions (or arbitrary length) built from the alphabet \mathbf{A} using square brackets and plus signs, and that the arrows in $\mathcal{F}_\times \mathbf{G}$ are parse trees for the arithmetic expressions.

3. Remarks

3.1. The relation between regular and context-free grammars, as defined above, is as follows. Given a regular grammar form a multigraph with an object $E_{X,Y}$ for each pair of objects X, Y of the reflexive graph and two types of arrows.

(i) For each triple of objects X, Y, Z of the reflexive graph, take an arrow $E_{X,Y}E_{Y,Z} \rightarrow E_{X,Z}$.

(ii) For each arrow of the reflexive graph from X to Y , take an arrow $1 \rightarrow E_{X,Y}$. Then label the arrows of the multigraph as follows: label arrows of type (i) with μ_2 , and the arrows of type (ii) by their label in the reflexive graph, except that the arrows labelled by identities in the reflexive graph are now labelled by μ_0 . The language defined by taking the object $E_{X,Y}$ in the resulting context-free grammar is the same as the language defined by the pair of objects X, Y of the regular grammar.

3.2. In the consideration of context-free languages it may sometimes be useful to consider an alphabet augmented by function symbols, rather than the alphabet of constants considered here.

Acknowledgment

I am grateful to Stefano Kasangian for discussions while he was visiting Sydney supported by the Australian Research Council, and to Phil Lavers, Stephen Hirst, Michael Johnson, and Bill Unger.

References

- [1] N. Chomsky and M.P. Schützenberger, The algebraic theory of context-free languages, in: Computer Programming and Formal Systems (North-Holland, Amsterdam, 1963) 118–161.
- [2] F. Lawvere, Qualitative distinctions between some toposes of generalized graphs, Preprint.
- [3] B. Pareigis, Categories and Functors (Academic Press, New York, 1970).