American Journal of Computational Linguistics *Microfiche 55 : 38*

Algebraic Parsing of Context-Free Languages

Stephen F. Weiss and Donald F. Stanat

Department of Computer Science
University of North Carolina
New West Hall 035A
Chapel Hill 27514

Abstract

A class of algebraic parsing techniques for context-free languages is presented. A grammar is used to characterize a parsing homomorphism which maps terminal strings to a polynomial semiring. The image of a string under an appropriate homomorphism contains terms which specify all derivations of the string. The work describes a spectrum of parsing techniques for each context-free grammar, ranging from a form of bottom-up to top-down procedures.

ALGEBRAIC PARSING OF CONTEXT-FREE LANGUAGES

I. Introduction

For many years syntactic analysis and the theory of formal languages have developed in a parallel, but not closely related, fashion. The work described here is an effort to relate these areas by applying the tools of formal power series to the problem of parsing.

This paper presents an algebraic technique for parsing a broad class of context-free grammars. By parsing we mean the process of determining whether a string of terminal symbols, χ, is a member of the language generated by grammar G (i.e., is $\chi \in L(G)$?) and, if it is, finding all derivations of χ from the starting symbol of G. We hope that posing the parsing problem in purely algebraic terms will provide a basis for examination and comparison of parsing algorithms and grammar classes.

Section II presents an overview of the algebraic parsing process. It provides a general notion of how the method works without going into detail. Section III contains the algebraic preliminaries and notational conventions needed in order to describe the parsing method precisely. The formal presentation of the parsing method and the proof of correctness form Section IV. Section V contains some interesting special cases of the theorem and presents some examples of parses.

II. Overview of the algebraic parsing process

The algebraic parsing formalism described here is applicable to all context-free grammars $G = \langle V_N, V_T, P, S \rangle$ except those that contain productions of the form $A \rightarrow B$ where A and B are both nonterminals, or erasing rules such as $A \rightarrow \epsilon$. The parsing process consists first of constructing (on the basis of the grammar G) a polynomial and a function defined on polynomials. A parse of χ is obtained by repeated applications of the function to a polynomial $P(\chi)$. The process has two features worthy of note. First, it produces all parses of χ in parallel. Second, the process of converting a grammar into the required algebraic form is straightforward and does not alter the structure of the grammar. This property, the preservation of grammatical structure, is particularly important in areas such as natural language analysis where the structure that a grammar provides is as important as the language it generates.

The polynomials we will use have terms of the form (Z,Δ), where Z is a string over an extended alphabet and Δ represents a sequence of productions of G. The process begins with a polynomial of ordered pairs representing χ, the string to be parsed. A function is repeatedly applied to the polynomial; the number of applications necessary is bounded by the input length. If the resulting polynomial contains a term (S,Δ) where S is the starting symbol in G, then Δ represents the production sequence used in generating χ from S. If no such pair occurs, then χ is not in L(G), and if multiple pairs

occur (S,Δ_1), (S,Δ_2), ... then χ is ambiguous and the Δ's specify the several parses. A precise formulation of the polynomial and the operations on it is given below.

III. Algebraic preliminaries and notation

A semigroup is formally defined as an ordered pair $\langle S,\cdot\rangle$ where S is a set (the carrier) and $\cdot$ is an associative binary operation. Similarly, a monoid is a triple consisting of a set, an operation and a two-sided identity (e.g., $\langle S,\cdot,1\rangle$). We will feel free to denote a monoid or semigroup by its carrier.

For any set V, V^* denotes the free monoid generated by V; $V^* = \langle V^*,\text{concatenation},\Lambda\rangle$. Similarly, V^+ denotes the free semigroup generated by V; $V^+ = \langle V^+,\text{concatenation}\rangle$. We denote the length of a string χ in V^* or V^+, by $|\chi|$.

For an arbitrary alphabet V, we define $\bar{V} = \{\bar{v}\,|\,v\in V\}$. The free half-group generated by V, H(V), is defined to be the monoid generated by $V \cup \bar{V}$ together with the relation $aa = 1$, where 1 is the monoid identity and a is any element of V. Note that in H(V) the elements of $\bar{V}$ are left inverses but not right inverses of the corresponding elements of V. We denote the extended alphabet $V \cup \bar{V}$ by Σ.

If $T = \langle T,\cdot,1\rangle$ and $Q = \langle Q,+,0\rangle$ are monoids, we denote by $T \times Q$ the product monoid $\langle T \times Q,\otimes,(1,0)\rangle$. The carrier of $T \times Q$ is the cartesian product $T \times Q$ and the operation $\otimes$ is defined to be the component-wise operation of T and Q:

$$(a,b) \otimes (c,d) = (a\cdot c,b+d).$$

A <u>semiring</u> is an algebraic system $\langle S,+,\cdot,0\rangle$ such that

$\langle S,+,0\rangle$ is a commutative monoid,

$\langle S,\cdot\rangle$ is a semigroup,

and the operation $\cdot$ distributes over +:

$$a\cdot(b+c) = a\cdot b + a\cdot c,$$

$$(a+b)\cdot c = a\cdot c + b\cdot c.$$

A <u>semiring</u> is commutative if the operation is commutative. A <u>semiring with identity</u> is a system $\langle S,+,\cdot,0,1\rangle$ where $\langle S,+,\cdot,0\rangle$ is a monoid. The semirings used in this paper are commutative and have identities. Furthermore, in each case the additive identity is a multiplicative zero:

$$0\cdot x = x\cdot 0 = 0.$$

The <u>boolean semiring B</u> consists of the carrier $\{0,1\}$ under the commutative operations + and $\cdot$, where $1\cdot 1 = 1+x = 1$ and $0+0 = 0\cdot x = 0$ for all $x \in \{0,1\}$.

For an arbitrary monoid M we denote by R(M) the semiring of polynomials described as follows:

1) Each <u>term</u> is of the form $c\alpha$ where $c \in B$ (the boolean semiring of coefficients) and $\alpha \in M$.

2) Each <u>polynomial</u> is a formula sum (under +) of a finite number of terms.

3) Addition and multiplication of terms is defined as follows:

   a) $b\alpha + c\alpha = (b + c)\,\alpha$

   b) $(b\alpha)(c\beta) = (bc)(\alpha\beta)$.

4) Addition and multiplication of polynomials is performed in the usual manner consistent with 3).

Note that all coefficients of R(M) are either 1 or 0. We will adopt the usual convention of not explicitly writing 1 for the terms with that coefficient and omitting terms with a coefficient of 0.

A context-free grammar is a system $G = \langle V_N, V_T, P, S \rangle$ where V_N and V_T are finite, disjoint, non empty sets denoted non-terminal and terminal symbols respectively. We denote by V the set $V_N \cup V_T$. The symbol S is the distinguished nonterminal from which all derivations begin, and P is the set of productions of G. A context-free grammer is proper if it does not contain productions of the form $A \rightarrow \epsilon$ (erasures) or $A \rightarrow B$ where A and B are both nonterminals.

It can easily be shown that the set of languages generated by proper context-free grammars is exactly the set of context-free languages. In addition, an arbitrary context-free grammar can be made proper by a straightforward method which alters the structure of the grammar very little. In this study we will deal with only proper context-free grammars. This guarantees that all terminal strings have a finite number of derivations in G, and thus makes possible our goal of finding all derivations of an input.

Productions of G will be indexed by integers. Thus $A \xrightarrow{i} M$ denotes that $A \rightarrow M$ is the i^{th} production in P. We will deal only with left-most derivations. A leftmost derivation is completely specified by the initial sentential form and the sequence of production indices. If $\Delta \in I^*$ is the sequence of production indices in the leftmost derivation of $N \in V^+$ from $M \in V^+$, we write $M \overset{\Delta}{\Longrightarrow} N$. The length of a derivation Δ is denoted by $|\Delta|$, and is equal to the number of production indices in Δ.

We will use, but not formally define, the notion of height of a

derivation', meaning the height of the corresponding derivation tree or the length of the longest path from the root to the frontier of the tree. The height of a derivation Δ will be denoted by $h(\Delta)$.

Since 'derivation' will always mean 'leftmost derivation' in the sequel, the following assertions hold:

Assertion 1: A derivation is of height 0 if and only if it is of length 0. A derivation is of height 1 if and only if it is of length 1.

Assertion 2: Let G be a proper context-free grammar, and

$$A \overset{\Delta}{\Longrightarrow} M$$

where $|\Delta|>0$. Then Δ is of height less than or equal to $|M|$.

Assertion 3: Let $G = \langle V_N, V_T, P, S\rangle$ be a context-free grammar, I an index set for P, and let the j^{th} production of G be

$$A \overset{j}{\rightarrow} a_1 a_2 \cdots a_m \qquad a_i \in V.$$

Let $j\Gamma$ be a derivation

$$A \overset{j\Gamma}{\Longrightarrow} M \qquad M \in V^+$$

of height $n + 1$. Then

$$\Gamma = \Delta_1 \Delta_2 \cdots \Delta_m \qquad \Delta_i \in I^*$$

and

$$M = M_1 M_2 \cdots M_m \qquad M_i \in V^+$$

and for all i, $1 \leq i \leq m$,

$$a_i \overset{\Delta_i}{\Longrightarrow} M_i$$

is a derivation of height n or less.

The algebraic structure used in this work is the semiring of polynomials $R(H \cdot I^*)$ where $H = H(V)$, the free half-group generated by V, and I is the index set of the set of productions P. We will use an initial segment of the natural numbers, {1,2,3,...}, as the index set I. Each term of a polynomial from $R(H \cdot I^*)$ consists of an element from $H \cdot I^*$ together with a coefficient from the boolean semiring B. The elements of $H \cdot I^*$ will be the basis for calculating the parses of a string λ. The elements of H will interact to determine if a product of terms characterizes a derivation. If so, the associated element of I^* is the sequence of production indices of the derivation.

The following notational conventions will be observed.

$G = \langle V_N, V_T, P, S \rangle$

$V = V_N \cup V_T$

$\Sigma = V \quad \bar{V}$

$Z \in \Sigma^+$

$z \in \Sigma$

$i, j, k \quad m, n \in \underline{N}$ (set of natural numbers).

$I \subseteq \underline{N}$

$\Delta, \Gamma, \theta \in I^*$

$\chi \in V_T^+$

$a, b, c \in V$

$A, B, C \quad V_N$

$M, N, P, Q \in V^*$

δ, g, Ψ, ν will denote functions. For the function g,

$$g^1(x) = g(x) \text{ and } g^k(x) = g(g^{k-1}(x)).$$

IV. An algebraic parsing theorem

Theorem (version 1): Let $G = \langle V_N, V_T, S, P\rangle$ be a proper context-free grammar. Then there exist homomorphisms ν, g, and δ,

$$\nu: V^* \to R(V \times I^*)^*$$

$$g: R(\Sigma \times I^*)^* \to R(\Sigma \times I^*)^*$$

$$\delta: R(\Sigma \times I^*)^* \to R(H \times I^*)$$

and a special polynomial $p \in R(\Sigma \times I^*)^*$ such that for every $\chi \in V_T^+$, $\chi = \chi_1 \cdots \chi_n$, $\chi_i \in V_T$,

$$\delta g^n\left[\prod_{i=1}^{n} p^n\ \nu(\chi_i)\right]$$

contains a term (S,Δ) if and only if Δ is a leftmost derivation of χ from S.

Construction for the proof:

Let $V = V_1 \cup V_2$ be an arbitrary exhaustive division of V:

$$V_1 \cup V_2 = V.$$

The construction is most economical when V_1 and V_2 are disjoint, but this is not required.

a.. $\nu: V^* \to R(V \times I^*)^*$

The function ν is the homomorphism induced by the following:

$\nu(a) = (a,\Lambda)$, $a \in V$ and Λ is the identity in I^*.

Since ν is a homomorphism, $\nu(\Lambda) = \Lambda$.

b. $g: R(\Sigma \times I^*)^* \rightarrow R(\Sigma \times I^*)^*$

The function g is the homomorphism induced by defining g on the generators of the domain as follows:

1. $g(\bar{a},\Delta) = (\bar{a},\Delta)$; $\bar{a} \in \bar{V}$, $\Delta \in I^*$

2i. $g(a,\Delta)$ contains the term (a,Δ); $a \in V$

2ii. If $A \rightarrow ab_1 \ldots b_n$ is the i^{th} production of P and $a \in V_1$ then $g(a,\Delta)$ contains $(A,i\Delta)(\bar{b}_n,\Lambda) \ldots (\bar{b}_1,\Lambda)$.

2iii. There are no other terms in $g(a,\Delta)$.

Note that because g is a homomorphism, $g(\Lambda) = \Lambda$, where Λ is the identity of the monoid $(\Sigma \times I^*)^*$

c. $\delta: R(\Sigma \times I^*)^* \rightarrow R(H \times I^*)$

The function δ is the canonical homomorphism which coalesces a product in $(\Sigma \times I^*)^*$ into a single ordered pair by component-wise multiplication of the first entries (thus allowing cancellation in H) and catenation of the second entries. For example,

$$\delta[(a,\Delta_1)(\bar{b},\Delta_2)(b,\Delta_3)(c,\Delta_4)] = (ac,\Delta_1\Delta_2\Delta_3\Delta_4).$$

d. The polynomial p is an element of $R(\Sigma \times I^*)^*$ defined as follows:

1. p contains the summand Λ;

2. If $a \in V_2$ and $A \rightarrow ab_1 \ldots b_n$ is the j^{th} production of P then p contains the summand $(A,j)(\bar{b}_n,\Lambda) \ldots (\bar{b}_1,\Lambda)(\bar{a},\Lambda)$.

3. p contains no other summands.

We adopt the convention that $p^k = \Lambda$ for $k \leq 0$.

Note that since p contains Λ, p^k contains Λ as well as all summands of p^j for $j \leq k$.

For notational convenience we adopt the following conventions. First, where no ambiguity can result, products in $R(\Sigma \times I^*)^*$ of the form

$$(z_1,\Delta_1)(z_2,\Delta_2) \ldots (z_n\Delta_n) \qquad z_i \in \Sigma,\ \Delta_i \in I^*$$

will be abbreviated as:

$$(z_1z_2 \ldots z_n,\ \Delta_1\Delta_2 \ldots \Delta_n).$$

No cancellation is implied by this notation since cancellation cannot occur in $R(\Sigma \times I^*)^*$. Second, we define the function Ψ_k as follows:

$$\Psi_k\colon V^* \to R(\Sigma \times I^*)^*$$

$$\Psi_k(a_1a_2 \ldots a_n) = \prod_{i=1}^{n} p^k\, \nu(a_i)$$

where $a_i \in V$ and p is the polynomial defined above. Note that, if $k \leq 0$, then $\Psi_k(a_1a_2 \ldots a_n) = \nu(a_1a_2 \ldots a_n)$, and $\Psi_k(\Lambda) = \Lambda$. Using this notation, we can re-state the theorem as follows:

Theorem (version 2): Let $G = \langle V_N, V_T, P, S\rangle$ be a proper context-free grammar. Then there exist maps Ψ, g and δ such that

$$\Psi\colon V^* \to R(\Sigma \times I^*)^*$$

$$g\colon R(\Sigma \times \perp^*)^* \to R(\Sigma \times I^*)^*$$

$$\delta\colon R(\Sigma \times I^*)^* \to R(H \times I^*)$$

such that for every $\chi \in V_T^+$, $\chi = \chi_1\chi_2 \cdots \chi_n$, $\chi_i \in V_T$, $\delta g^n \Psi_n(\chi)$ contains a term (S,Λ) if and only if $S \overset{\Delta}{\Longrightarrow} \chi$.

The proof of the theorem rests on three lemmas. Lemma I implies the "if" part of the theorem; Lemma III implies the "only if" part. Lemma II is used in the proof of Lemma III.

Lemma I: Let $M \in V^+$, $A \in V$, and $A \overset{\Delta}{\Longrightarrow} M$. Then for all $k > h(\Delta)$, $\delta g^k \Psi_k(m)$ contains (A,Δ).

Proof (by induction on $h(\Lambda)$, the height of the derivation Δ):

Basis: If $h(\Delta) = 0$, then $\Delta = \Lambda$ and $M = A$. Then $\Psi_k(A) = p^k(A,\Lambda)$. Since Λ is a summand of p, it follows that (A,Λ) is a summand of $p^k(A,\Lambda)$, and therefore (A,Λ) is a summand of $\delta g^k \Psi_k(A,\Lambda)$. Thus the derivation $A \overset{\Lambda}{\Longrightarrow} A$ is represented in $\delta g^k \Psi_k(A)$ by (A,Λ), which establishes the basis.

Induction: Let Δ be a derivation of height $n + 1$, $A \overset{\Delta}{\Longrightarrow} M$. By assertion 3,

$$\Delta = j\Gamma_1\Gamma_2 \cdots \Gamma_r$$

$$M = M_1M_2 \cdots M_r$$

where

$$A \overset{j}{\rightarrow} a_1a_2 \cdots a_r$$

and

$$a_i \overset{\Gamma_i}{\Longrightarrow} M_i$$

where $h(\Gamma_i)$ n.

Since $\delta g^{k}\psi_{k}(M) = \delta g^{k}\psi_{k}(M_1)\delta g^{k}\psi_{k}(M_2) \ldots \delta g^{k}\psi_{k}(M_r)$, if $k \geq n$ then by the induction hypothesis, $\delta g^{k}\psi_{k}(M_j)$ contains the summand (a_j,Γ_j). Consider the term of $g^{k}\psi_{k}(M_1)$ which cancels to (a_1,Γ_1) in $R(H \times I^*)$. This term must be of the form $(a_1,\Gamma_1')T$, where Γ_1' is a prefix of Γ_1. Either $a_1 \in V_1$ or $a_1 \in V_2$. The sum $\delta g^{k+1}\psi_{k+1}(M_1)$ contains $\delta g g^{k}\psi_{k}(M_1)$, which contains $\delta g(a_1,\Gamma_1')T$. If $a_1 \in V_1$, then $g(a_1,\Gamma_1')$ contains $(\overline{Aa_2a_3\ldots a_r}, j\Gamma_1')$, and $\delta g(a_1,\Gamma_1')T$ contains $(\overline{Aa_2a_3\ldots a_r}, j\Gamma_1)$. On the other hand, the sum $\delta g^{k+1}\psi_{k+1}(M_1)$ also contains $\delta p g^{k}\psi_{k}(M_1)$. If $a_i \in V_2$, then $(\overline{Aa_1a_2\ldots a_r}, j)$ is a summand of p, and therefore $\delta p(a_1\Gamma_1')T$ contains $(\overline{Aa_2a_3\ldots a_r}, j\Gamma_1)$. Thus in either case, $\delta g^{k+1}\psi_{k+1}(M_1)$ contains the summand $(\overline{Aa_2a_3\ldots a_r}, j\Gamma_1)$ and since every summand of $\delta g^{k}\psi_{k}(M_j)$ is a summand of $\delta g^{k+1}\psi_{k+1}(M_i)$, it follows that $\delta g^{k+1}\psi_{k+1}(M)$ contains

$$(\overline{Aa_2a_3\ldots a_r}, j\Gamma_1)(a_2,\Gamma_2)(a_3,\Gamma_3)\ldots(a_r,\Gamma_r)$$

$$= (A, j\Gamma_1\Gamma_2\ldots\Gamma_r) = (A,\Delta).$$

This completes the proof.

<u>Lemma II</u>: Let $a \in V$, $\Gamma \in I^*$. For $k \geq 0$, all terms of $g^{k}(a,\Gamma)$ are of the form $(b,\Delta\Gamma)(\bar{c}_m,\Lambda)\ldots(\bar{c}_1,\Lambda)$ where $b \in V$, $\bar{c}_i \in \bar{V}$, $m \geq 0$, $\Delta \in I^*$ and $b \overset{\Delta}{\Longrightarrow} ac_1\ldots c_m$.

For notational convenience we abbreviate $c_1\ldots c_m$ by N. Hence we denote $(b,\Delta\Gamma)(\bar{c}_m,\Lambda)\ldots(\bar{c}_1,\Lambda)$ by $(bN,\Delta\Gamma)$.

Proof by induction on k, the number of applications of g. By definition, $g^{0}(a,\Gamma) = (a,\Gamma)$ which establishes the assertion for the

value k = 0.

Assume the assertion holds for k < n and consider $g^{n+1}(a,\Gamma) = gg^{n}(a,\Gamma)$. By the induction hypothesis, all terms of $g^{n}(a,\Gamma)$ are of the form $(b\bar{N},\theta\Gamma)$ where $b \overset{\theta}{\Longrightarrow} aN$. Hence terms of $g^{n+1}(a,\Gamma)$ are of the form $g(b\bar{N},\theta\Gamma)$. Since g limited to $\bar{V}$ is the identity, $g(b\bar{N},\theta\Gamma) = [g(b,\theta\Gamma)](\bar{N},\Lambda)$. By definition of g, $g(b,\theta\Gamma)$ contains only terms of the form $(C\bar{M},j\theta\Gamma)$ where $C \overset{j}{\rightarrow} bM$ is a production. Therefore terms of $g^{n+1}(a,\Gamma)$ are of the form

$$(C\bar{M},j\theta\Gamma)(\bar{N},\Lambda) = (C\overline{\bar{M}\bar{N}},j\theta\Gamma)$$

and since $C \overset{j}{\rightarrow} bM$ and $b \overset{\theta}{\Longrightarrow} aN$ it follows that $C \overset{j\theta}{\Longrightarrow} aNM$.

Corollary: All terms of $g^{k}(a\bar{M},\Gamma)$ are of the form $(b\overline{NM},\Delta\Gamma)$.

Lemma III: If $\delta g^{k}\Psi_{k}(M)$ contains $(A\bar{N},\Delta)$, then $A \overset{\Delta}{\Longrightarrow} MN$.

Proof by induction on the length of M:

Basis: Let $a \in V$ and assume

$$\delta g^{k}\Psi_{k}(a) \text{ contains } (A\bar{N},\Delta).$$

If p_i represents an arbitrary summand of p other than Λ, then every term of $g^{k}\Psi_{k}(a)$ can be represented in the form

$$\prod_{i=1}^{n} g^{k}(p_i)\ g^{k}(a,\Lambda)$$

where $0 \leq n \leq k$ and n denotes the number of nontrivial summands of p which are factors of the term.

By construction, every summand of p is either $\Lambda\cdot$ or of the form

$$(B_i\bar{P}_i, j_i) \text{ where } B_i \in V_N,\ P_i \in V^+,\ j_i \in I$$

and $B_i \overset{j_i}{\rightarrow} P_i$ is a production in G.

By Lemma. II. every term of $g^k(B_i\bar{P}_i, j_i)$ is of the form:

$$(C_i\bar{M}_i\bar{P}_i, \Gamma_i j_i) \text{ where } C_i \in V_N,\ M_i,\ P_i \in V^*,\ \Gamma_i \in I^*$$

and $C_i \overset{\Gamma_i}{\Longrightarrow} B_iM_i$;

By the same lemma, it follows that every term of $g^k(a,\Lambda)$ is of the form

$$(C_{n+1}\bar{M}_{n+1}, \Gamma_{n+1}) \text{ where } C_{n+1} \in V,\ M_{n+1} \in V^*,\ \Gamma_{n+1} \in I^*$$

and $C_{n+1} \overset{\Gamma_{n+1}}{\Longrightarrow} M_{n+1}$.

Hence every term of $g^k\Psi_k(a)$ is of the form

$$\left[\prod_{i=1}^{n} (C_i\bar{M}_i\bar{P}_i, \Gamma_i j_i) \;\middle|\; (C_{n+1}M_{n+1}, \Gamma_{n+1}) \quad 0 < n \leq k \right]$$

where $C_i \overset{\Gamma_i j_i}{\Longrightarrow} P_iM_i$ for $1 \leq i \leq n$ and $C_{n+1} \overset{I_{n+1}}{\Longrightarrow} M_{n+1}$ (1)

By assumption there is a term t of $g^k\Psi_k(a)$ such that $\delta[t] = (\Lambda\bar{N}, \Lambda)$; t must be in the form indicated above. In order for t to cancel under δ, the following must be true:

$C_1 = A$ since C_1 cannot cancel from t,

$\bar{P}_i = \bar{Q}_i\bar{C}_{i+1}$ for $1 < i \leq n$ since $C_2 \ldots C_{n+1}$ must all cancel from t.

Therefore

$$t = \left[\prod_{i=1}^{n} (C_i \bar{M}_i \bar{Q}_i \bar{C}_{i+1}, \Gamma_i j_i) \right] (C_{n+1} \bar{M}_{n+1}, \Gamma_{n+1}).$$

This cancels to $(A\bar{N}, \Delta)$ as required with

$$A = C_1$$

$$N = M_{n+1} Q_n M_n Q_{n-1} M_{n-1} \cdots Q_1 M_1$$

$$\Delta = \Gamma_1 j_1 \Gamma_2 j_2 \cdots \Gamma_n j_n \Gamma_{n+1}.$$

Then by (1),

$$C_i \overset{\Gamma_i j_i}{\Longrightarrow} C_{i+1} Q_i M_i, \quad 1 \leq i \leq n, \text{ and}$$

$$C_{n+1} \overset{\Gamma_{n+1}}{\Longrightarrow} M_{n+1}$$

Hence, since $C_1 = A$,

$$A \overset{\Gamma_1 j_1 \Gamma_2 j_2 \cdots \Gamma_n j_n \Gamma_{n+1}}{\Longrightarrow} M_{n+1} Q_n M_n Q_{n-1} M_{n-1} \cdots Q_1 M_1$$

and thus

$$A \overset{\Delta}{\Longrightarrow} N.$$

This establishes the basis.

Induction: Assume that for all $M \in V^*$ such that $|M| \leq n$, if $\delta g^k \Psi_k(M)$ contains $(A\bar{N}, \Delta)$ then $A \overset{\Delta}{\Longrightarrow} MN$. Let $\hat{M} = Ma$ be a string such that $|Ma| = n+1$ and $\delta g^k \Psi_k(Ma)$ contains $(A\bar{N}, \Delta)$. Because δ g and Ψ are homomorphisms,

$$\delta g^k \Psi_k(Ma) = [\delta g^k \Psi_k(M)][\delta g^k \Psi_k(a)].$$

Then $\delta g^k \psi_k(M)$ must contain a term (T_1, Δ_1) and $\delta g^k \psi_k(a)$ must contain a term (T_2, Δ_2) such that $T_1 T_2 = A\bar{N}$ and $\Delta = \Delta_1 \Delta_2$.

In order for this to occur, T_2 must be of the form $(B\bar{N}_2)$ where $B \in V$, $N_2 \in V^*$, and T_1 just be of the form $(A\bar{N}_1\bar{B})$ where $A \in V$, $N_1 \in V^*$, and $\bar{N} = \bar{N}_1\bar{N}_2$. (If T_1 and T_2 were not of this form, cancellation to $A\bar{N}$ would be impossible.) Thus $\delta g^k \psi_k(M)$ contains $(A\overline{N_1 B}, \Delta_1)$, and by the induction hypothesis

$$A \overset{\Delta_1}{\Longrightarrow} MBN_1.$$

Also $\delta g^k \psi_k(a)$ contains $(B\bar{N}_2, \Delta_2)$ and by the basis

$$B \overset{\Delta_2}{\Longrightarrow} aN_2.$$

It follows that

$$A \overset{\Delta_1\Delta_2}{\Longrightarrow} MaN_2N_1$$

and since $\hat{M} = Ma$ and $N = N_2N_1$,

$$A \overset{\Delta}{\Longrightarrow} \hat{M}N$$

which completes the proof.

The theorem now follows from Lemmas I and III and Assertion 2. The 'if' part follows from Lemma I and Assertion 2, and the 'only if' part follows immediately from Lemma III for the special case of $N = \Lambda$.

As we have stated the theorem, the length of χ is used to determine a sufficient number of applications of g and ψ. Alternatively, the theorem could be formulated in terms of the heights of derivations

of χ; if Δ is a derivation of χ of height k, then for every $n \geq k$, the term (S,Δ) will be in the polynomial $\delta g^n \Psi_n(\chi)$. Furthermore, it follows from Lemma III that no harm is done by choosing the value of n too large, i.e., no 'false' derivation terms will occur.

In the first statement of the theorem, the derivation terms are obtained from the polynomial $\delta g^n \prod_{i=1}^{n} p^n \nu(\chi_i)$ which can be re-written in the form

$$\delta \prod_{i=1}^{n} g^n [p^n \nu(\chi_i)]$$

Although we have used a constant value of n (equal to the length of χ) for both the powers of the map g and the polynomial p, some economy can be gained in this respect. In fact, the powers of g and p can decrease from left to right so long as they remain large enough to perform the appropriate computations on the suffix strings of χ. Thus, the theorem is true (but considerably more difficult to prove) if one instead uses a parsing polynomial of the form

$$\delta \prod_{i=1}^{n} g^{n-i+1} [p^{n-i+1} \nu(\chi_i)].$$

V. Special cases of the theorem

A number of interesting special cases occur based on the choice of V_1 and V_2.

Case 1. $V_1 = V_T$.

$V_2 = V_N$.

The function g handles all productions of the form

$$A \rightarrow \alpha M \qquad \alpha \in V_T, \; M \in V^*,$$

while p handles productions of the form

$$A \to BM \qquad B \in V_N,\ M \in V^*.$$

Notice that since g is nontrivial on only V_T, g need be used only once; i.e.,

$$g^k(\alpha,\Gamma) = g(\alpha,\Gamma) \qquad k > 1.$$

The parsing polynomial is then

$$\delta\{g[\Psi_k(\chi)]\}.$$

The special case of $V_1 = V_T$ and $V_2 = V_N$ results in a particularly simple form if the grammar is in Greibach no mal form. The polynomial $p = (\Lambda,\Lambda)$ and therefore has no effect. Since g need only be applied once, all derivations are found in one step.

Example 1:

$$G = \langle\{S,A,B\},\ \{a,b\},\ S,\ P\rangle$$

$P =$ 1. $S \to a\Lambda$

2. $A \to AB$

3. $A \to A$ $\qquad V_1 = \{a,b\}$

4. $B \to b$ $\qquad V_2 = \{S,A,B\}$

$$p = (\Lambda,\Lambda) + (A,2)(\bar{B},\Lambda)(\bar{A},\Lambda)$$

$$g(a,\Lambda) = (a,\Lambda) + (S,1)(\bar{A},\Lambda) + (A,3)$$

$$g(b,\Lambda) = (b,\Lambda) + (B,4).$$

For the string χ = aabb, the parsing polynomial $g[\Psi_k(\chi)]$ then contains (among other things) for all $k \geq 2$,

$$g(a,\Lambda)\ p^2\ g(a,\Lambda)\ g(b,\Lambda)\ g(b,\Lambda).$$

This contains:

$$[(S,1)(\bar{A},\Lambda)][(A,2)(\bar{B},\Lambda)(\bar{A},\Lambda)(A,2)(\bar{B},\Lambda)(\bar{A},\Lambda)][(A,3)][(B,4)][(B,4)].$$

Applying δ we get

$$(S,122344).$$

Case 2. $V_1 = V.$

$V_2 = \phi.$

The entire job of parsing is now done by g, since the polynomial p is equal to (Λ,Λ). Hence the parsing polynomial is

$$\delta[g^k(\chi,\Lambda)].$$

Example 2: We use the same grammar and input string as above.

$V_1 = \{S, A, B, a, b\}.$

$V_2 = \phi.$

$g(S,\Lambda) = (S,\Lambda)$

$g(A,\Lambda) = (A,\Lambda) + (A,2)(\bar{B},\Lambda)$

$g(B,\Lambda) = (B,\Lambda)$

$g(a,\Lambda) = (a,\Lambda) + (S,1)(\bar{A},\Lambda) + (A,3)$

$g(b,\Lambda) = (A,\Lambda) + (B,4).$

The parsing polynomial for aabb is

$$g^k(a,\Lambda)\ g^k(a,\Lambda)\ g^k(b,\Lambda)\ g^k(b,\Lambda).$$

For $k \geq 3$, this contains

$$[g^1(a,\Lambda)][g^3(a,\Lambda)][g^1(b,\Lambda)][g^1(b,\Lambda)]$$

which in turn contains

$[(S,1)(\bar{A},\Lambda)][g^2(A,3)][(B,4)][(B,4)]$ after one application of g,

$[(S,1)(\bar{A},\Lambda)][g^1(A,23)(\bar{B},\Lambda)][(B,4)][(B,4)]$ after two; and

$[(S,1)(\bar{A},\Lambda)][(A,223)(\bar{B},\Lambda)(\bar{B},\Lambda)][(B,4)][(B,4)]$ after three.

Applying δ results in (S,122344) as before.

Case 3. $V_1 = \phi$.

$V_2 = V$.

Now the entire parse is handled by p. The parsing polynomial becomes

$$\delta[\Psi_k(\chi)].$$

VI. Observations

The major theorem presented here shows how context-free parsing may be carried out by purely algebraic means. All parses of an input string are developed in parallel and the process is guaranteed to terminate. As we have described the process, the number of terms of a parsing polynomial for a string $\chi \in V_T^+$ is unreasonably large. However, most of the terms in such a polynomial are not associated with a derivation in the grammar, and methods exist for reducing the computation by disregarding dead-end terms before they are completely evaluated. By applying such techniques in a straightforward fashion, and choosing V_1 and V_2 in various ways,

the algebraic method can be associated in natural ways with classical parsing techniques. For example, the algebraic process in case 1 above is a goal directed top-down approach similar to the predictive analyzer. Case 2 is the algebraic version of generalized bottom-up.

Parsing algorithms are typically so different one from another that they are incomparable. But using techniques described above, many parsing algorithms may be posed in a single algebraic framework. This may facilitate the comparison and evaluation of parsers and of various classes of grammars.

REFERENCES

Chomsky, N. and M. Schutzenberger (1963), The Algebraic Theory of Context-Free Languages, in "Computer Programming and Formal Systems". (P. Braffort and D. Hirschbert, Eds.), North Holland, Amsterdam.

Ginsburg, S. and H. G. Rice (1963), Two Families of Languages Related to ALGOL, JACM 9, pp. 350-371.

Shamir, Eliahu (1967), A Representation Theorem for Algebraic and Context-Free Power Series in Non-Commuting Variables, Information and Control 11, pp. 239-254

Stanat, D. F. (1972), Approximation of Weighted Type 0 Languages by Formal Power Series, Information and Control 21, pp 344-381.

Stanat, D. F. (1972), A Homomorphism Theorem for Weighted Context-Free Grammars, J. Comput. System Sci. 6, pp. 217-232

Weiss, S. F., D. F. Stanat and G. A. Mago (1973), Algebraic Parsing Techniques for Context-Free Grammars, in "Automata, Languages and Programming" (M. Nivot, Ed.), pp. 493-498, North Holland/American Elsevier.