GENERALISED GRAMMARS AND HOMOMORPHIC IMAGES OF RECOGNIZABLE SETS

by

T.S.E. Maibaum

Research Report CS-73-30

Department of Applied Analysis and
Computer Science

University of Waterloo

Waterloo, Ontario, Canada

November, 1973

GENERALISED GRAMMARS
AND
HOMOMORPHIC IMAGES OF RECOGNIZABLE SETS

T.S.E.Maibaum

Queen Mary College
and
Royal Holloway College
University of London

1973

"You see it's like a portmanteau - there are two meanings packed into one word."

The Walrus and the Carpenter.

"... in nature as in mathematics every form is content for 'higher' forms and every content form of what it 'contains'."

Structuralism.

ABSTRACT

A new setting for the study of formal languages is introduced: many-sorted alphabets (operator domains). The derived algebra and the completion (algebra) of a many-sorted algebra are defined and used to study the derived operations of that algebra. Semi-Thue productions and grammars over many-sorted alphabets are introduced and the classes of regular, context free, monotonic, and context sensitive grammars are defined by restricting the types of productions allowed. The concepts of concatenation, complex product, Kleene closure, projection and homomorphism are generalised.

Finite automata over many-sorted alphabets are introduced and the classical results of the conventional theory generalised (among them the equivalence of the classes of deterministic and non-deterministic finite automata, the solution to the emptiness, finiteness, and equivalence problems, etc.). Regular grammars, the class of regular sets, and equational systems are studied and a theorem proved relating the classes of sets of terms defined by each method.

Context free grammars are shown to have canonic modes of derivation. A normal form is introduced. The class of sets generated by context free grammars is shown to be closed under the operations of complex product, Kleene closure, non-deterministic linear finite state transformation, intersection with recognizable sets, projection and inverse projection.

The Fundamental Theorem is then proved. It shows that the class of context free sets over a many-sorted alphabet is equal to the class of sets which are the homomorphic images of recognizable sets over a simply related alphabet.

The classical Substitution Theorem is proved in this new setting and the usual 'yield' theorems are extended. The closure of the class of recognizable (context free) sets under the operation of homomorphism is shown. A grammar-independent definition of languages is introduced.

ACKNOWLEDGEMENTS

Many thanks to Professor P.J.Landin and Dr.J.W.Thatcher for their advice and encouragement. Also to Raymond Turner and Alex Dubinsky for many helpful discussions. Most of this research was done while the author was a Tutorial Research Student at Royal Holloway College.

CONTENTS

Page

(0) For Jan

(1) Anyunak s apunak

0 INTRODUCTION

0.0 Motivation

This work is a report arising directly out of a process of generalisation which began with the algebraic approach to conventional formal language and automata theory taken by Büchi and Wright. It was continued, independently in Doner and in Thatcher and Wright, by generalising the concepts of the conventional theory to the so-called generalised theory: that is, generalising from the study of sets of strings to the study of sets of terms. Sets of terms were defined to be the subsets of the carrier of the word algebra (totally free algebra) on a finite operator domain (variously ranked alphabet, stratified alphabet).

It was shown that the 'derivation trees' of context free sets of strings could be characterised as recognizable sets over the appropriate operator domain (see Mezei and Wright, Brainerd (1), Thatcher (1)). The natural questions to ask at this point were: Can the concept of context free set (and grammar) be generalised from the conventional (string) case to define context free sets of terms (and grammars) over a finite operator domain? If so, can we characterise the 'derivation trees' of context free sets of terms by means of recognizable sets in, perhaps, a more complicated operator domain? A subsidiary question is: What class of 'derivation trees' of sets of strings is characterised by the class of context free sets of terms (if these latter are definable). (A short discussion of the first two questions with P.J.Landin was the original motivation for this work.)

Let us analyse the significant difference between regular and context free string grammars which allowed us to answer the first question in the affirmative. Consider a production in a regular grammar, say $A \to aB$ (A, B non-terminals and a terminal), and a production in a context free grammar, say $A \rightarrow w$ (A a non-terminal and w a string of terminals and non-terminals in any order). Note that, in a derivation in a regular grammar, a non-terminal can appear only at the end of a string (because of the form of the productions) and so we can substitute a string (derivable from that non-terminal) only at the end of another string. On the other hand, in a derivation in a context free grammar, a non-terminal may appear anywhere in a string (again because of the form of the productions) and so we can substitute a string (derivable from that non-terminal) in the middle of a string. In generalising this situation to terms and term grammars, the analogy seems to be that, in a regular term grammar, non-terminals may appear only as individual symbols (nullaries) in a term (or, more picturesquely, on the leaves of trees) whereas, in context free term grammars, non-terminals may appear as symbols of any rank in a term (on the leaves or as internal nodes of a tree). Regular term grammars (or, more correctly, a slight generalisation of them) were studied in Brainerd (1). Context free term grammars were presented, independently, in Rounds (1), (2) but were arrived at through a very different motivation.

As to the subsidiary question, it turns out that the 'derivation trees' of indexed sets of strings can be characterised by context free sets of terms. This also appeared independently in Rounds (1), (2).

The second question above (motivated by the relationship between context free sets of strings and recognizable sets of terms) also has an affirmative answer but involved us in the extension of formal language and automata theory concepts one step further: studying subsets of the carrier of the word algebra (totally free algebra) over a many-sorted alphabet. These many-sorted alphabets are used to define many-sorted algebras, the latter being a simple extension of the usual notion of algebras where, instead of having one set as the carrier of the algebra and operations on that set, a many sorted algebra has several sets (of different sorts) and operations on them. Many-sorted algebras (and structures) are not new in Computer Science and have infrequently appeared in the literature (see Kaplan, Engeler). A simple example is the case of the arithmetic operations in ALGOL which are defined only for certain sorts of numbers (as, for instance '+' which is defined if the operands are both complex and gives a complex number as a result; or if one operand is an integer while the other is real and gives a real as a result; etc.).

It turns out, not unexpectedly, that the usual concepts of regular and context free grammars, recognizable sets, equational sets, regular sets, finite automata, etc., generalise in a straightforward manner to the many-sorted case. One may well ask at this point whether this added generality is really worthwhile? We believe it is for the following reasons: (a) It allows us to present answers to what we consider to be the very important questions discussed above; (b) It seems a 'fixed point' in the process of generalisation (conventional to generalised to many-sorted) in the sense that, in order to characterise the 'derivation trees' of context free sets of terms over a many-sorted operator domain, it is not necessary to generalise again to a concept more complicated than 'many-sortedness'; (c) 'Many-sorted structure' seems to be a concept which is widely used in Computer Science (as in the example above, the use and manipulation of different types of data structures, etc.).

The analysis presented allows us to draw the conventional and generalised theories as special cases of the many-sorted theory. Many concepts are clarified in this new setting and a number of anomalies removed. We have reason to believe that many-sorted alphabets are a 'natural' setting for the study of formal languages and automata.

0.1. Summary

In Chapter I we present the definition of many-sorted structures and study their properties. Many-sorted algebras appeared independently in the literature as 'algebras with a scheme of operators' (Higgins) and later as 'heterogeneous algebras' (Birkhoff and Lipson) in essentially equivalent forms. The present study uses a notation suggested by J.W.Thatcher which is similar to the usual notation for algebras and which we hope is simple and clear. In Section 1 we present some basic set theory. Section 2 introduces structures. In Section 3, we study algebras (a special case of structures) as they will be the setting for most of the remaining work. The Fundamental Theorem of Algebra is stated to serve as the basis of much of what follows. Two new concepts are introduced in Section 4: completions of algebras (after an idea in Thatcher (2)) and derived algebras. These algebras systematize the study of the so-called derived operations of an algebra (the operations definable in terms of the operations of an algebra). In this chapter, only the results of the last section are original.

Chapter II is devoted to the presentation and study of formal language concepts in the many-sorted case. The setting is provided by semi-Thue productions and grammars which have been generalised from the conventional case. (A different definition for the generalised case was suggested in Thatcher (1)). The classes of regular, context free, monotonic, and context sensitive grammars are defined as semi-Thue grammars with certain restrictions on the types of productions allowed. In the Remark at the end of the chapter, the conventional theory is shown to be a special case of the many-sorted theory. In this chapter, the definitions of semi-Thue, context sensitive, and monotonic grammars and homomorphism

with respect to concatenation are original. For the remainder of the chapter, only the setting is new.

The definitions of deterministic and non-deterministic finite automata are given in the first section of Chapter III. We then present the well known results concerning finite automata and the sets they recognize (among them the equivalence of the classes of deterministic and non-deterministic finite automata, the relationship between congruences and recognizability, the closure of the recognizable sets under the Boolean operations, and the solutions of the emptyness, finiteness and equivalence problems for finite automata). In Section 2 we prove that the class of sets generated by regular grammars is the same as the class of recognizable sets. The former class is also shown to be closed under the operations of complex product, Kleene closure, projection, and inverse projection. Sections 3 and 4 are devoted to briefly stating the theories of regular sets and equational sets, respectively. The Equivalence Theorem at the end of the chapter relates these four classes of sets. We believe this is the first time such a theorem has been stated for sets of terms. Otherwise, only the setting is really original in the chapter (although some of the proofs are done in a new way).

The class of context free grammars is studied in Chapter IV. We first prove the existence of a canonic mode of derivation (the so-called outside-in mode). Then the equivalence of the sets generated by a grammar using only outside-in derivations and using unrestricted derivations is shown. We next prove a normal form result which seems a direct generalisation of the Chomsky normal form for context free string grammars. The class of context free sets of terms is shown to be closed under the operations of intersection with regular sets, complex product, Kleene closure, projection, and inverse projection. The work presented in this chapter is original, although we refer to the literature for proofs which are detailed and unilluminating and can be easily generalised to the many sorted case.

Chapter V is the core of this work. It contains the Fundamental Theorem which characterises the 'derivation trees' of context free sets of terms as recognizable sets over a derived alphabet. The concept of context free set is extended to cover sets of derived operations of the word algebra. This chapter is wholly original.

Chapter VI details some of the consequences of the Fundamental Theorem. In Section 1 we generalise complex product and Kleene closure (the generalisation of Chapter II being partial) and state the classic Substitution Theorem. The next section proves a number of equivalences: between the classes of derivation trees of indexed sets of strings and context free sets of terms and between the classes of the 'derivation trees of the derivation trees' of indexed sets of strings and recognizable sets of terms over certain alphabets. Motivated by the latter equivalence, we define indexed sets of terms and state the Second Fundamental Theorem (showing the equivalence of the classes of the 'derivation trees of the derivation trees' of indexed sets of terms and recognizable sets of terms over certain related alphabets). Section 3 contains the proof of the closure of the classes of context free sets of terms and recognizable sets of terms under the operation of homomorphism with respect to composition. The last section gives a definition of the classes of context free sets and indexed sets in terms of congruences (and independently of grammars). This chapter is original work.

The last chapter sums up the work and presents some consequences and open problems suggested by this theory.

0.2. Notational Conventions

The reader is referred to Halmos for the foundations of set theory and to Hopcroft and Ullman (among many others) for the foundations of conventional formal language theory. The following notation will be used throughout (new notations being introduced where we need them):

$\{a_o,\ldots,a_{n-1}\}$	The set with elements $a_o,\ldots,a_{n-1}$.
$a \in A$	a is an element of the set A.
$A \sqsubseteq B$	The set A is a subset of the set B.
$A \sqcup B$	The union of the sets A and B.
$A \sqcap B$	The intersection of the sets A and B.
$A - B$	The complement of the set B in the set A.
$A \times B$	The cross-product of the set A and the set B.
$\underline{N}$	The set of natural numbers.
$\langle a_o,\ldots,a_{n-1}\rangle$ or $(a_o,\ldots,a_{n-1})$	An ordered set (n-tuple) with n elements.
Σ^*	The set of strings on the set Σ.
$w(\in \Sigma^*)$	A string over the set $\Sigma(w = w_o \ldots w_{n-1})$.
$\ell(w)$	The length of the string w.
$\lambda(\in \Sigma^*)$	The empty string (string of length 0).
Σ^+	$\Sigma^* - \{\lambda\}$.

I PRELIMINARIES

I.0 Introduction

This chapter introduces the algebraic concepts and definitions we need for our explication. They are by no means intended to be in any way complete and references are given where appropriate. The reader who is unfamiliar with any of the material is advised to consult the literature. The above is not intended to imply that all the material in this chapter can be found elsewhere and where it cannot an effort has been made to be as complete as possible. (References: Halmos, Birkhoff, Cohn (1), Grätzer (1), Higgins, Birkhoff and Lipson.)

I.1 Sets

Given two sets A and B, a relation between A and B is a triple $\langle A, \Phi, B \rangle$ where $\Phi \subseteq A \times B$. We will write such a triple as $\Phi: A \to B$ and, where the sets A and B are obvious from the context, name the relation by Φ. We will generally use upper case Greek letters to denote relations. The set of such relations on the sets A and B will be denoted by $[A \to B]$.

Each relation $\Phi: A \to B$ defines in a unique way the relation $p\Phi: pA \to pB$, where, for $A' \subseteq A$, $p\Phi(A') = \{y \in B | (x, y) \in \Phi \text{ for some } x \in A'\}$. Again, if $A' \subseteq A$, define $\Phi|_{A'}: A' \to B$ by $\Phi|_{A'} = \{(x, y) \in \Phi \text{ and } x \in A'\}$. $\Phi|_{A'}$ is called the restriction of Φ to A'.

Every relation $\Phi: A \to B$ has the converse $\Phi^{-1}: B \to A$ defined by $\Phi^{-1} = \{(x, y) | (y, x) \in \Phi\}$.

The relation $\Delta_A: A \to A$ defined by $\Delta_A = \{(x,x) | x \in A\}$ is called the diagonal or identity in A.

A relation $\Phi: A \to A$ is said to be symmetric if $\Phi = \Phi^{-1}$, antisymmetric if $\Phi \cap \Phi^{-1} \subseteq \Delta_A$, and reflexive if $\Phi \supseteq \Delta_A$.

For any relations $\Phi:A \to B$ and $\Psi:B \to C$ we define the composition $\Psi\circ\Phi:A \to C$ with $\Psi\circ\Phi = \{(x, y) \in A \times C \mid (x, z) \in \Phi$ and $(z, y) \in \Psi$ for some $z \in B\}$. A relation $\Phi:A \to A$ is said to be transitive if $\Phi\circ\Phi \subseteq \Phi$.

A relation $\phi:S \to T$ is said to be a function if $\phi \subseteq S \times T$ has the properties:

(i) If $(x, y) \in \phi$ and $(x, y') \in \phi$, then $y = y'$; and

(ii) For each $x \in S$, there exists a $y \in T$ such that $(x, y) \in \phi$.

The sets S and T are called the source and target, respectively, of ϕ. If the second of the above two conditions does not hold, then $\phi:S \to T$ is said to be a partial function. We will generally use lower case Greek letters to denote functions and use $\phi(x) = y$ to indicate $(x, y) \in \phi$. The set of all functions $\phi:S \to T$ will be denoted by $(S \to T)$.

A function $\phi:A \to B$ is said to be surjective or onto if $\phi\circ\phi^{-1} = \Delta_B$, injective or one-one if $\phi^{-1}\circ\phi = \Delta_A$, and bijective if it is both onto and one-one.

Let A and I be any sets. Then the set $\{\phi(i) \in A \mid \phi:I \to A$ and $i \in I\}$ is called a family of elements of A, indexed by I. If $x_i = \phi(i)$ for $i \in I$, then the family is denoted by $\{x_i\}_{i\in I}$ (or, when I is known, by x or $\{x_i\}$). x_i is called the i-coordinate of $\{x_i\}$ and I is called the index set. Every set can be indexed (by itself, for example).

An equivalence (relation) on a set A is a relation $\Phi:A \to A$ which is reflexive, symmetric, and transitive. That is, we have

(i) $\Delta_A \subseteq \Phi$;

(ii) $\Phi^{-1} = \Phi$;

and (iii) $\Phi\circ\Phi \subseteq \Phi$.

We will generally use the lower case Roman letters q, r, s to denote equivalences. If q is any equivalence on A, then for each $x \in A$ we define a subset x^q of A, the q-class of x, by $x^q = \{y \in A | (x, y) \in q\}$. It is obvious that $x^q = y^q$ if and only if $(x, y) \in q$.

A partition on a set A is a function $\rho_I : I \to pA$ for some set I, such that:

(i) for $i, j \in I$ and $i \neq j$, $\rho_I(i) \cap \rho_I(j) = \phi$

and (ii) $\bigcup_{i \in I} \rho_I(i) = A$ (where $\bigcup_{i \in I} A_i = \{x | x \in A_i \text{ for some } i \in I\}$ for $\{A_i\}_{i \in I}$) .

That is, the sets $\rho_I(i)$ are pairwise disjoint and the union of all such sets is the set A. The sets $\rho_I(i)$ are called the classes of the partition. Given a partition $\rho_I : I \to pA$, we can define an equivalence q_{ρ_I} on A by the following: $(x, y) \in q_{\rho_I}$ if and only if $x, y \in \rho_I(i)$, some $i \in I$. Conversely given an equivalence q on A, we can define a partition $\rho_{\{x^q\}} : \{x^q\} \to pA$ on A by $\rho_{\{x^q\}}(x^q) = x^q$. Thus there is a one-one correspondence between partitions on a set A and equivalences on A.

Let $q : A \to A$ be an equivalence on the set A. Let A/q be the subset of pA consisting of all q-classes of A. A/q is called the quotient set of A (mod q). We define the function $\hat{q} : A \to A/q$, called the natural function from A to A/q, by $\hat{q}(x) = x^q$. We note that $\hat{q}$ is surjective.

I.2 Many-sorted Structures

Let I be any set, called the set of sorts. A many-sorted alphabet sorted by I, is an indexed family of sets $\Sigma = \{\Sigma_{\langle w,i \rangle}\}_{\langle w,i \rangle \in I^* \times I}$. $\Sigma_{\langle w,i \rangle}$ is the set of (relation) symbols of type $\langle w, i \rangle$ and if $f \in \Sigma_{\langle w,i \rangle}$, then we say that f has type $\langle w, i \rangle$, sort i, arity w, and rank $\ell(w)+1$. A symbol of type $\langle \lambda, i \rangle$ is called an individual or constant symbol of sort i.

Let $M = \{M_i\}_{i \varepsilon I}$ be any family of sets indexed by I and Σ a many-sorted alphabet. A (many-sorted) relational structure over the alphabet Σ, or more briefly, a Σ-structure, is a non-empty family of sets M together with a family of functions $\alpha_{\langle w,i \rangle} : \Sigma_{\langle w,i \rangle} \rightarrow [M^w \rightarrow M_i]$. That is, to each $f \varepsilon \Sigma_{\langle w,i \rangle}$, we assign a relation $f_M : M^w \rightarrow M_i$, where $f_M = \alpha_{\langle w,i \rangle}(f)$ and, for $w = w_o \ldots w_{n-1}$, $M^w = M_{w_o} \times \ldots \times M_{w_{n-1}}$. The family of sets M with such a structure is denoted by M_Σ (or M if it is obvious which many-sorted alphabet we are using). The underlying family of sets M is also called the carrier of M_Σ. Let $\hat{M} = \bigcup_{i \varepsilon I} M_i$.

Let M_Σ and N_Σ be two Σ-structures. We say that N_Σ is a substructure of M_Σ if $N_i \subseteq M_i$ for all $i \varepsilon I$ and $f_N = f_M \cap (N^w \times N_i)$ for each $f \varepsilon \Sigma_{\langle w, i \rangle}$, all $\langle w,i \rangle \varepsilon I^* \times I$. That is, if the carrier of N_Σ is a 'subset' of the carrier of M_Σ and if the relations f_N of N_Σ are the restrictions to N of f_M. The cross-product $M_\Sigma \times N_\Sigma$ of two Σ-structures M_Σ and N_Σ is defined as the structure with carrier $M \times N = \{M_i \times N_i\}$ and relations defined component-wise. That is, for $c_o, \ldots, c_{n-1} \varepsilon \widehat{M \times N}$, $(c_o, \ldots, c_{n-1}) \varepsilon f_{M \times N}$ if and only if $(a_o, \ldots, a_n) \varepsilon f_M$ and $(b_o, \ldots, b_n) \varepsilon f_N$ where $(a_j, b_j) = c_j$ $(0 \leq j \leq n)$.

Given Σ-structures M_Σ and N_Σ, an indexed family of functions $\{\phi_i : M_i \rightarrow N_i\}$ is called a homomorphism of M_Σ into N_Σ if $\{\phi_i\}$ is a substructure of $M_\Sigma \times N_\Sigma$. Equivalently, $\{\phi_i : M_i \rightarrow N_i\}$ is called a homomorphism if, for $(a_o, \ldots, a_n) \varepsilon M^w \times M_i$, $(a_o, \ldots, a_n) \varepsilon f_M$ implies $(\phi_{w_o}(a_o), \ldots, \phi_{w_n}(a_{w_n})) \varepsilon f_N$. We usually write the homomorphism as $\phi : M_\Sigma \rightarrow N_\Sigma$. A homomorphism $\phi : M_\Sigma \rightarrow N_\Sigma$ is said to be an epimorphism if each $\phi_i : M_i \rightarrow N_i$ is onto. It is called a monomorphism if each $\phi_i : M_i \rightarrow N_i$ is one-one. It is called an isomorphism if each $\phi_i : M_i \rightarrow N_i$ has an inverse. The inverse homomorphism is denoted by $\phi^{-1} : N_\Sigma \rightarrow M_\Sigma$.

A <u>congruence</u> on the structure M_Σ is an indexed family $\{q_i : M_i \to M_i\}_{i\varepsilon I}$ of equivalences (denoted by $q:M \to M$) which has the property that q is a substructure of $M \times M$. Using the indexed family of functions $\hat{q} = \{\hat{q}_i : M_i \to M_i/q_i\}$, we define the <u>quotient structure</u> M/q as the structure with carrier M/q and relations $f_{M/q}$ defined as follows: For $(b_o, \ldots, b_n) \varepsilon (M/q)^w \times (M/q)_i$, $(b_o, \ldots, b_n) \varepsilon f_{M/q}$ if and only if there exists $(a_o, \ldots, a_n) \varepsilon M^w \times M_i$ such that $\hat{q}_{w_j}(a_j) = b_j$ $(0 \leq j \leq n)$ and $(a_o, \ldots, a_n) \varepsilon f_M$. $\hat{q}:M \to M/q$ is obviously an epimorphism. Given a homomorphism $\phi:M \to N$, a congruence $q_\phi : M \to M$ is induced on M. This congruence is called the <u>kernel</u> of ϕ and $q_\phi = \phi^{-1} \circ \phi$.

A many-sorted alphabet Σ is said to be <u>finite</u> if the disjoint union of the family of sets Σ is a finite set. A many-sorted Σ-structure M_Σ is <u>finite</u> if Σ is a finite alphabet and if $\hat{M}$ $(= \bigsqcup_{i\varepsilon I} M_i)$ is a finite set.

<u>Example 1</u>: Let $I = \{E, V\}$ and Σ be the many-sorted alphabet with $\Sigma_{\langle VV,E \rangle} = \{\uparrow\}$ and all other $\Sigma_{\langle w,i \rangle} = \phi$. Then a graph with vertices (nodes) G_V and directed edges G_E, together with a relation $\uparrow_G$ (with $(v_1, v_2, e) \varepsilon \uparrow_G$ interpreted to mean that e is an edge from v_1 to v_2), is a many-sorted structure. Let $H_V \sqsubseteq G_V$ and $H_E = \{e \varepsilon G_E |$ there exist $v_1, v_2 \varepsilon G_V$ such that $(v_1, v_2, e) \varepsilon \uparrow_G\}$. Then $\{H_V, H_E\}$ is a substructure of G. Consider the functions $\phi_E : H_E \to G_E$ and $\phi_V : H_V \to G_V$ defined by $\phi_E(e) = e$ (for $e \varepsilon H_E$) and $\phi_V(v) = v$ (for $v \varepsilon H_V$), respectively. $\phi = \{\phi_V, \phi_E\}$ is easily seen to be a homomorphism (in fact it is a monomorphism).

<u>Example 2</u>: If I consists of a single element, say i, and each relation symbol f is assigned a (non-empty) relation in the only possible way (f of type $\langle ii \ldots i, i \rangle$) then the Σ-structures are just the structures of the usual theory.

I.3. Many-sorted Algebras

Let A_Σ be a Σ-structure, Σ a many-sorted alphabet sorted by I. Recall that A_Σ is an indexed family of sets $\{A_i\}$ together with an indexed family of functions $\alpha_{\langle w,i\rangle} : \Sigma_{\langle w,i\rangle} \to [M^w \to M_i]$. If it happens that the family of functions α is such that each $\alpha_{\langle w,i\rangle} : \Sigma_{\langle w,i\rangle} \to (M^w \to M_i)$, that is $\alpha_{\langle w,i\rangle}$ assigns to each $f \in \Sigma_{\langle w,i\rangle}$ a function from M^w to M_i, then A_Σ is called a <u>many-sorted Σ-algebra</u>. In this section we will study the properties of Σ-algebras, as they provide the setting for most of the remainder of this work.

We begin by stating that $f \in \Sigma_{\langle w,i\rangle}$ is called an <u>operator (symbol)</u> of <u>type</u> $\langle w,i\rangle$, <u>sort</u> i, <u>arity</u> w, and <u>rank</u> $\ell(w)$ (note the slight difference between this definition and that for relation symbols in the previous section). If in the previous section (less the first paragraph) we replace the symbols '[' and ']' by '('and')' respectively, and the word 'structure' by the word algebra, then we have a series of definitions and properties valid for algebras. Since for each $f \in \Sigma_{\langle w,i\rangle}$, f_A is a function, we write $f_A(a_o, \ldots, a_{n-1}) = a_n$ instead of $(a_o, \ldots, a_n) \in f_A$, for $(a_o, \ldots, a_n) \in A^w \times A_i$.

<u>Example 1</u>: Let $I = \{S, \Delta, \Omega\}$. S is the sort 'states', Δ is the sort 'input symbols', and Ω is the sort 'output symbols'. Let $A = \{A_i\}_{i \in I}$ be a family of non-void sets such that A_Ω is finite. A_S is called the set of states, A_Δ the set of input symbols, A_Ω the set of output symbols. Let $\Sigma_{\langle S\Delta,S\rangle} = \{\delta\}$ and $\Sigma_{\langle S\Delta,\Omega\rangle} = \{\xi\}$ while all other $\Sigma_{\langle w,i\rangle} = \phi$. Now let M be a sequential machine (with states A_S, input symbols A_Δ, and output symbols A_Ω) and assign to δ the change of state operation of M and to ξ the output function of M. We have thus characterised the sequential machine M as the many-sorted algebra A.

Analogously, we can describe a finite automaton as a Σ'-algebra B, with sorting set $I = \{S, \Delta\}$, where B_S is a finite set of states and B_Δ is a finite input alphabet. Σ' consists of the set $\Sigma'_{\langle S\Delta,S\rangle} = \{M\}$, called the next state function, and the set $\Sigma'_{\langle \lambda,S\rangle} = \{q_o\}$, called the initial state of B, with all other $\Sigma'_{\langle w,i\rangle} = \phi$.

Example 2: Let $I = \{M, S\}$ and $\Sigma_{\langle MM,M\rangle} = \{*\}$, $\Sigma_{\langle \lambda,M\rangle} = \{1\}$, and $\Sigma_{\langle MS,S\rangle} = \{.\}$ with all other $\Sigma_{\langle w,i\rangle} = \phi$. Let A_M be the carrier of a monoid and $*_A$, 1_A the monoid binary operation, identity element respectively. Let A_S be any set and assign to $. \in \Sigma_{\langle MS,S\rangle}$ any operation $._A : A_M \times A_S \to A_S$ which satisfies the following laws for any $m, n \in A_M$ and $s \in A_S$:

(i) $1_A ._A s = s$

and (ii) $(m *_A n) ._A s = m ._A (n ._A s)$.

The Σ-algebra A then describes the action of a monoid on a set.

Example 3: Let C be any abstract category and $|C|$ the objects of C. then $I = |C| \times |C|$, $\Sigma_{\langle \lambda,(B,B)\rangle} = \{\varepsilon_B\}$ for each $(B, B) \in I$, and $\Sigma_{\langle (B,C)(C,D),(B,D)\rangle} = \{o\}$ for all (B,C), $(C,D) \in I$ (with all other $\Sigma_{\langle w,i\rangle} = \phi$). For each $(B, C) \in I$, let $A_{(B,C)} = \pi(B, C)$, the set of morphisms from A to B. For each individual symbol $\varepsilon_B \in \Sigma_{\langle \lambda,(B,B)\rangle}$, let $(\varepsilon_B)_A$ be the unique identity element of $\pi(B,B)$. For the operation symbol $o \in \Sigma_{\langle (B,C)(C,D),(B,D)\rangle}$, let o_A be the composition of morphisms from $\pi(B, C)$ and $\pi(C, D)$. The axioms for a category can then be stated as identities in this Σ-algebra.

Example 4: Let I consist of a single element, say i, and let Σ be an alphabet sorted by I. If each operation symbol $f \in \Sigma$ is assigned a non-empty operation in the only possible way (f of type $\langle i...i,i\rangle$), then the

Σ-algebras are just the algebras of the usual theory.

Let Σ be a many-sorted alphabet and $X = \{X_i\}_{i \varepsilon I}$ be any family of sets indexed by I. The family of Σ-**expressions**, $W_\Sigma(X) = \{W_\Sigma(X)_i\}_{i\varepsilon I}$, indexed by I is the least family of subsets of

$\{(\bigcup_{\langle w,i\rangle \varepsilon I^* \times I} \Sigma_{\langle w,i\rangle}) \bigcup (\bigcup_{i\varepsilon I} X_i)\}^*$ satisfying:

(o) $X_i \bigcup \Sigma_{\langle \lambda,i\rangle} \sqsubseteq W_\Sigma(X)_i$ for each $i \varepsilon I$;

and (i) for each $f \varepsilon \Sigma_{\langle w,i\rangle}$ $(w = w_o \ldots w_{n-1}, n > 0)$ and any n-tuple

$(t_o, \ldots, t_{n-1}) \varepsilon W_\Sigma(X)^w$, $ft_o \ldots t_{n-1} \varepsilon W_\Sigma(X)_i$.

We can impose a Σ-algebra structure on the family of sets $W_\Sigma(X)$ by associating with each $f \varepsilon \Sigma_{\langle w,i\rangle}$ the operation defined by: If $(t_o, \ldots, t_{n-1}) \varepsilon W_\Sigma(X)^w$, then $f_{W_\Sigma(X)}(t_o, \ldots, t_{n-1}) = ft_o \ldots t_{n-1}$. This Σ-algebra is called the <u>Σ-word algebra</u> on the generators X. If each $X_i = \phi$, we denote $W_\Sigma(\{\phi\}_{i\varepsilon I})$ by W_Σ. If Y is another indexed family of sets then we write $W_\Sigma(X, Y)$ to denote the algebra $W_\Sigma(\{X_i \bigcup Y_i\}_{i\varepsilon I})$.

<u>Example 5</u>: Let $I = \{0, 1\}$, $\Sigma_{\langle \lambda,o\rangle} = \{\lambda\}$, $\Sigma_{\langle \lambda,1\rangle} = \{a, b\}$, $\Sigma_{\langle 10,0\rangle} = \{*\}$, and $\Sigma_{\langle 11,1\rangle} = \{+\}$. Let $X_1 = \{x\}$ and $X_o = \phi$. Then we may describe the set $W_\Sigma(X)_o$ as <u>terms</u> <u>(words)</u> of the form λ, $*a\lambda$, $*b\lambda$, $*x\lambda$, $*{+}aa\lambda$, $*{+}ax\lambda$, etc. (Generally, terms in $W_\Sigma(X)_o$ are of the form *+( )+( ).)

Elements of $W_\Sigma(X)_1$ are of the form a, b, x, +aa, +bb, +ab, +ba, +ax, +xa, +bx, +xb, ++aa+aa, ++aa+bb, etc. (Generally, terms in $W_\Sigma(X)_1$ are of the form ++( )+( ).)

If we change our example by specifying that $X_o = X_1 = \phi$, then $(W_\Sigma)_o$ and $(W_\Sigma)_1$ are as above but without terms involving the generator x.

Let M_Σ be a Σ-structure. We define a Σ-structure $(pM)_\Sigma$ on $pM = \{pM_i\}_{i\varepsilon I}$ as follows: Given $(A_o, \ldots, A_{n-1}, A_n) \varepsilon (pM)^W \times pM_i$ and $f \varepsilon \Sigma_{\langle w,i \rangle}$, then $(A_o, \ldots, A_n) \varepsilon f_{pM}$ if and only if $A_n = \{a_n \varepsilon M_i | \exists a_o \ldots \exists a_{n-1}[(a_o,\ldots,a_n)\varepsilon f_M \text{ and } a_j \varepsilon M_{w_j} \text{ for } 0 \leq j \leq n-1]\}$. It is obvious that, in fact, $(pM)_\Sigma$ is a Σ-algebra and so we may write $f_{pM}(A_o, \ldots, A_{n-1}) = A_n$. We call $(pM)_\Sigma$ the raised algebra of M_Σ. Given two Σ-structures M_Σ and N_Σ, it is obvious that any homomorphism $\phi:M_\Sigma \to N_\Sigma$ induces in a unique way a homomorphism $p\phi:(pM)_\Sigma \to (pN)_\Sigma$.

The following theorem is of fundamental importance in the study of algebras and is stated without proof. (The interested reader is referred to Cohn (1), Grätzer (1), Birkhoff.)

Fundamental Theorem of Algebra: Given a Σ-algebra A_Σ and generators $X = \{X_i\}_{i\varepsilon I}$, then any indexed family of functions $\{\psi_i:X_i \to A_i\}$ (usually written $\psi:X \to A$) extends in a unique way to a homomorphism $\overline{\psi}:W_\Sigma(X) \to A_\Sigma$. In particular, if each $X_i = \phi$, there is a unique homomorphism from W_Σ to A_Σ.

Example 6: Let Σ be the alphabet of the previous example. Let A be a Σ-algebra where $A_o = (\Sigma_{\langle \lambda,1 \rangle} \cup \{c\})^*$ and $A_1 = (\Sigma_{\langle \lambda,1 \rangle} \cup \{c\})^+$. To $+ \varepsilon \Sigma_{\langle 11,1 \rangle}$ we assign the operation of concatenation $+_A$, defined if both arguments of $+_A$ are not the empty string. To $* \varepsilon \Sigma_{\langle 10,0 \rangle}$ we assign the operation of concatenation $*_A$, defined only if the first argument is not the empty string. Consider the assignment $\phi_1:x \to c$. This extends in a unique way to a homomorphism $\overline{\phi}:W_\Sigma(X) \to A$ (X as in the previous example). Intuitively, we can see this homomorphism as being the function 'display the term $t \varepsilon \hat{W}_\Sigma(X)$ as a tree in the usual way and read the names of the leaf nodes of the tree from left to right, reading c for x'. (See Thatcher (1) for a similar idea called 'frontier function'.) We note that the string

$c \notin A_o$ and $\lambda \notin A_1$. If we let $B_o = \Sigma^*_{\langle\lambda,1\rangle}$ and $B_1 = \Sigma^+_{\langle\lambda,1\rangle}$ then there is a unique homomorphism $\psi: W_\Sigma \to B$. The significance of this example will become clear in Chapter VI.

In the work that follows, we will often use the convention 'f ε Σ' to mean 'f ε $\Sigma_{\langle w,i\rangle}$, some $\langle w, i\rangle$ ε $I^* \times I$'. Although this usage is technically incorrect, it is nevertheless convenient and unambiguous. Also, in most of the work that follows it will not be necessary to make a distinction between an element f ε Σ, and its associated relation (function) f_M in a structure (algebra) M. We will denote both by f and it will be simple to decide from the context which we mean. If this is not the case we will make the distinction explicit. This procedure is quite common in modern algebra and should not cause any problems. (For example, in groups the unit operator of many groups is denoted by the same symbol '1' and the group operator by the same symbol '·' (or sometimes '+')).

I.4. Derived Operations, Completions of Algebras and Derived Algebras

We begin this section with an example of a Σ-word algebra which will play an important role in our development.

Example 1 (Thatcher (2)): Let Σ be a many-sorted alphabet and w ε I^*. Consider the set $X_w = \{x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}\}$, where $w = w_o \ldots w_{n-1}$. We say X_w is indexed by w ε I^*. As an I-sorted alphabet $(X_w)_i = \{x_{j,i} | j < n\}$ for each i ε I. Thus, our definition of word algebra with a family of generators applies to the alphabet Σ and generators $\{(X_w)_i\}$ and we can easily find $W_\Sigma(\{(X_w)_i\}_{i\varepsilon I})$. We will denote this algebra by $W_\Sigma(X_w)$. A term in $W_\Sigma(X_w)_i$ will be said to be of type $\langle w, i\rangle$.

Let $V \subseteq I^*$ and let $X_V = \{X_w\}_{w\varepsilon V}$. Again we can easily make X_V into a family of generators sorted by I and define the algebra $W_\Sigma(X_V)$ as above.

Let A be a Σ-algebra. Consider the following question. What can we say about the 'operations on the family of sets A definable from the operations of the algebra A?' (See also Cohn (1), Grätzer (1), Benabou, Lawvere (1).)

To expedite our work, consider the term (also called <u>Σ-expression</u>) $e = f(g(x_1), x_o)$ where $I = \{0, 1\}$; $\Sigma_{\langle oo,o\rangle} = \{f\}$, $\Sigma_{\langle 1,o\rangle} = \{g\}$, $\Sigma_{\langle 11,0\rangle} = \{h\}$, $\Sigma_{\langle \lambda,1\rangle} = \{a\}$; and $X_o = \{x_o, y_o\}$, $X_1 = \{x_1, y_1\}$. Then $W_\Sigma(X)_o = \{x_o, y_o, h(x_1, a), f(x_o, g(x_1))$, etc.$\}$ while $W_\Sigma(X)_1 = \{x_1, y_1, a\}$. We can define a function, called a derived operation, $v_e:A_1 \times A_o \to A_o$ by $v_e(b_1, b_o) = \overline{\phi}(f(g(x_1), x_o))$ where $\overline{\phi}$ is the unique homomorphism $\overline{\phi}:W_\Sigma(X) \to A$ generated by the assignments $\phi_o(x_o) = b_o$, $\phi_1(x_1) = b_1$ (and any assignments to y_o and y_1 consistent with the sorting). On the other hand, we could associate with e the function $v_e':A_o \times A_1 \to A_o$ defined by $v_e'(b_o, b_1) = v_e(b_1, b_o)$. How can we distinguish between these two functions and which should we associate with the expression e? (This problem arises because of the fact that $A^w \approx A^{w'}$ (A^w is isomorphic to $A^{w'}$) whenever, for $w = w_o \dots w_{n-1}$ and $w' = w_o' \dots w_{n-1}'$ $(w_o', \dots, w_{n-1}')$ is a permutation of $(w_o, \dots, w_{n-1})$).

We accomplish this by introducing a special set of variables, namely that of Example 1, so that our notation will reflect exactly which operation we intend to denote by any expression. So, given a Σ-algebra A and an expression e in $W_\Sigma(X_w)_i$, some Σ, $i \in I$, $w \in I^*$, we will define a <u>derived operation</u>, v_e, of type $\langle w,i\rangle$ as follows: For all $(a_o,\dots,a_{n-1}) \in A^w$, $v_e(a_o, \dots, a_{n-1}) = \overline{\phi}(e)$ where $\overline{\phi}$ is the unique homomorphism $\overline{\phi}:W_\Sigma(X_w) \to A$ generated by the assignments $\phi_{w_j}(x_{j,w_j}) = a_j$ for $0 \leq j \leq n-1$.

Returning to our example above, let $w = 10$, $v = 01$ and $X_w = \{x_{o,1}, x_{1,o}\}$, $X_v = \{x_{o,o}, x_{1,1}\}$. Then $f(g(x_{o,1}), x_{1,o}) \in W_\Sigma(X_w)_o$

corresponds to the first derived operation $v_e : A_1 \times A_o \to A_o$ and $f(g(x_{1,1}), x_{o,o}) \in W_\Sigma(X_v)_o$ corresponds to the second derived operation $v_e' : A_o \times A_1 \to A_o$. Also note that $e = f(g(x_{o,1}), x_{1,o}) \in W_\Sigma(X_u)_o$, where $u = 100$ and $X_u = \{x_{o,1}, x_{1,o}, x_{2,o}\}$, corresponds to an operation $v_e : A_1 \times A_o \times A_o \to A_o$ which has a dummy third argument. Thus we see that the same expression in two sets $W_\Sigma(X_w)_i$ and $W_\Sigma(X_{w'})_i$ defines two distinct operations.

Let Σ be a many-sorted alphabet and $V \subseteq I^*$. Consider the new I-sorted alphabet Σ' where $\Sigma'_{\langle w,i\rangle} = W_\Sigma(X_w)_i$ for each $w \in V$, $i \in I$. That is, the operator symbols of type $\langle w,i\rangle$ in Σ' are enumerated by the expressions in $W_\Sigma(X_w)_i$. We make a Σ-algebra A into a Σ'-algebra by having $e \in \Sigma'_{\langle w,i\rangle}$ name the operation $v_e : A^w \to A_i$, as defined above. Let $V_o = \{w \in I^* | \Sigma_{\langle w,i\rangle} \neq \phi$ for some $i \in I\}$. If $V_o \subseteq V$, then we call the Σ'-algebra, obtained from A_Σ, a <u>completion</u> of A and denote it by $\overline{A}(V)$. Its operations of type $\langle w,i\rangle$, $w \in V$, are the derived operations of A of type $\langle w,i\rangle$. The definition of completion ensures that the operations of A are also operations of $\overline{A}(V)$ because if $f \in \Sigma_{\langle w,i\rangle}$, then the operation f_A of A is exactly v_e where $e = f(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}})$. If $V = V_o$, we call $\overline{A}(V_o)$ the <u>initial completion</u> of A and denote it by $\overline{A}()$. We note that $\overline{A}()$ is the 'least' completion of A which one can obtain if the operations of A are also to be operations of a completion of A. If $V = I^*$, we call $\overline{A}(I^*)$ the <u>full completion</u> of A and denote it by $\overline{A}$. $\overline{A}$ is the 'largest' completion of A in the sense that any operation of a completion of A is also an operation of the full completion of A.

Now, consider some completion $\overline{A}(V)$ of a Σ-algebra A. Let $e \in \Sigma'_{\langle w,i\rangle}$ name some derived operation $v_e : A^w \to A_i$ in A as defined above. Also, let $e_j \in \Sigma'_{\langle v,w_j\rangle}$ for $0 \leq j \leq n-1$. Then it is obvious that we can define a derived operation of type $\langle v, i \rangle$ on A as the operation named by $Sub_w(e; e_o, \ldots, e_{n-1})$, where $\overline{\sigma} = Sub_w : W_\Sigma(X_w) \to W_\Sigma(X_v)$ is the homomorphism generated by the assignments $\sigma(x_{j,w_j}) = e_j$ for $0 \leq j \leq n-1$. (This is the familiar substitution operation (Thatcher (1), (2) and Rounds (2)) where some expression is substituted for each occurrence of a given variable in another expression, simultaneous substitution for different variables being allowed.) We call any such instance of the substitution operator composition (as we are using it to compose functions) and denote it by $c_{\langle w,v,i \rangle}$. What can we say about the operations in the completion $\overline{A}(V)$ of the algebra A under such operations of composition? It seems obvious that it forms a many-sorted algebra closely related to $\overline{A}(V)$ (or to A).

We proceed to clarify this relationship with the following definitions: Let A be a Σ-algebra (sorted by I) and $\overline{A}(V)$ some completion of A. Define $D_V(A)$, the derived algebra of A, with respect to $V \subseteq I^*$, to be the many-sorted algebra with:

(i) sorting set $D_V(I) \subseteq I^* \times I$ where $D_V(I) = \{\langle w,i \rangle \mid w \in V,\ i \in I\}$;

(ii) an indexed family of operator symbols $D_V(\Sigma)$ called the derived alphabet, where:

(a) $f \in \Sigma_{\langle w,i \rangle}$ (f of type $\langle w,i \rangle$ in Σ) is an individual symbol of type $\langle \lambda, \langle w,i \rangle\rangle$ in $D_V(\Sigma)$. That is, $f \in D_V(\Sigma)_{\langle \lambda, \langle w,i \rangle\rangle}$;

(b) δ is an individual symbol of type $\langle \lambda, \langle w,w_j \rangle\rangle$ for each $w \in V$ and each $0 \leq j \leq n-1$ $(w = w_o \ldots w_{n-1})$. These operators are called projections (and their significance will be clarified in later chapters). We usually indicate that $\delta \in D_V(\Sigma)_{\langle \lambda, \langle w,w_j \rangle\rangle}$ by explicitly writing δ_w^{j+1};

(c) c is an operator symbol of type $\langle\langle w,i \rangle\langle v,w_o \rangle \ldots \langle v,w_{n-1} \rangle, \langle v,i \rangle\rangle$ (arity $\langle w,i \rangle\langle v,w_o \rangle \ldots \langle v,w_{n-1} \rangle$, sort $\langle v,i \rangle$, rank $n+1 (= \ell(w)+1)$) for each $\langle w,v,i \rangle \in I^+ \times I^* \times I$ such that $w,v \in V$. We usually write $c_{\langle w,v,i \rangle}$ for $c \in D_V(\Sigma)_{\langle\langle w,i \rangle\langle v,w_o \rangle \ldots \langle v,w_{n-1} \rangle, \langle v,i \rangle\rangle}$;

(iii) the indexed family of sets $D_V(A) = \{D_V(A)_{\langle w,i\rangle}\}_{\langle w,i\rangle \in V \times I}$ as carrier (where each $D_V(A)_{\langle w,i\rangle}$ is, of course, the set of derived operations of A of type $\langle w,i\rangle$); and

(iv) an indexed family of assignments

$$\alpha_{\langle w,i\rangle} : D_V(\Sigma)_{\langle w,i\rangle} \to (D_V(A)^w \to D_V(A)_i),$$

where $\langle w,i\rangle \in D_V(I)^* \times D_V(I)$, such that

(a) $f \in D_V(\Sigma)_{\langle \lambda, \langle w,i\rangle\rangle}$, $\langle w,i\rangle \in V \times I$, is assigned the derived operation f_A;

(b) $\delta_w^{j+1} \in D_V(\Sigma)_{\langle \lambda, \langle w,w_j\rangle\rangle}$, $w \in V$, is assigned the operation of projection;

and (c) $c_{\langle w,v,i\rangle} \in D_V(\Sigma)_{\langle\langle w,i\rangle\langle v,w_o\rangle \ldots \langle v,w_{n-1}\rangle, \langle v,i\rangle\rangle}$, $v,w \in V$, is assigned the operation of composition.

We denote the derived algebras of A with respect to I^* and V_o by $D(A)$ and $D_{()}(A)$, respectively. Also, denote by $D(I)$ and $D(\Sigma)$ the sorting set and alphabet, respectively, of $D(A)$; by $D_{()}(I)$ and $D_{()}(\Sigma)$ the sorting set and alphabet, respectively, of $D_{()}(A)$. Note that $D(A)$ is an example of what is usually called a clone of A (Hall, Cohn (1), Benabou, Lawvere (1)).

An important point to note immediately is that if Σ is finite then so is $D_{()}(\Sigma)$. If A is a Σ-algebra with Σ finite (and so I can be taken to be finite as well), then $D_{()}(A)$ is a $D_{()}(\Sigma)$-algebra with both $D_{()}(\Sigma)$ and $D_{()}(I)$ finite. Also note that each $A_i \in A$, some $i \in I$, is also an element of the family of sets $D_V(A)$ and is sorted in $D_V(A)$ by $\langle \lambda, i\rangle \in D_V(I)$.

Example 2: Let $I = \{0, 1\}$, $\Omega_{\langle\lambda,0\rangle} = \{\lambda\}$, $\Omega_{\langle\lambda,1\rangle} = \{a\}$, $\Omega_{\langle 10,0\rangle} = \{*\}$, $\Omega_{\langle 11,1\rangle} = \{+\}$ and all other $\Omega_{\langle w,i\rangle} = \phi$. The derived algebra $D_{()}(W_\Omega)$ of the algebra W_Ω is obtained as follows:

(i) The sorting set is $D_{()}(I) = \{ \langle \lambda,0 \rangle, \langle \lambda,1 \rangle, \langle 10,0 \rangle, \langle 10,1 \rangle, \langle 11,0 \rangle, \langle 11,1 \rangle \}$;

(ii) The indexed set of operator symbols is

$D_{()}(\Omega)_{\langle \lambda, \langle \lambda,0 \rangle\rangle} = \{\lambda\}$, $D_{()}(\Omega)_{\langle \lambda, \langle \lambda,1 \rangle\rangle} = \{a\}$,

$D_{()}(\Omega)_{\langle \lambda, \langle 10,0 \rangle\rangle} = \{\delta^2_{10}, *\}$, $D_{()}(\Omega)_{\langle \lambda, \langle 10,1 \rangle\rangle} = \{\delta^1_{10}\}$,

$D_{()}(\Omega)_{\langle \lambda, \langle 11,0 \rangle\rangle} = \phi$, $D_{()}(\Omega)_{\langle \lambda, \langle 11,1 \rangle\rangle} = \{\delta^1_{11}, \delta^2_{11}, +\}$ and

$c \in D_{()}(\Omega)_{\langle\langle w,i\rangle\langle v,w_o\rangle \ldots \langle v,w_{n-1}\rangle, \langle v,i\rangle\rangle}$ for each $(w,v,i) \in \{10, 11\} \times \{\lambda, 10, 11\} \times I$ (with all other $D_{()}(\Omega)_{\langle w,i\rangle} = \phi$);

(iii) Let $X_o = \{y_{1,0}\}$ and $X_1 = \{x_{0,1}, x_{1,1}, y_{0,1}\}$ be a family of generators indexed by I. Then the element of the carrier of $D_{()}(W_\Omega)$ of sort $\langle \lambda, 0 \rangle$ is the set $(W_\Omega)_o$, the element of sort $\langle \lambda, 1 \rangle$ is $(W_\Omega)_1$, the element of sort $\langle 10, 0 \rangle$ is $W_\Omega(\{y_{0,1}, y_{1,0}\})_0$, the element of sort $\langle 10, 1 \rangle$ is $W_\Omega(\{y_{0,1}, y_{1,0}\})_1$, the element of sort $\langle 11, 0 \rangle$ is $W_\Omega(\{x_{0,1}, x_{1,1}\})_0$ and the element of sort $\langle 11, 1 \rangle$ is $W_\Omega(\{x_{0,1}, x_{1,1}\})_1$;

(iv) λ, a, *, + name the constants λ, a, $*y_{0,1}y_{1,0}$, $+ x_{0,1}x_{1,1}$ respectively. $c_{\langle w,v,i \rangle} \in D_{()}(\Omega)_{\langle\langle w,i\rangle\langle v,w_o\rangle \ldots \langle v,w_{n-1}\rangle, \langle v,i \rangle\rangle}$ is assigned the operation of composition described previously. δ^{j+1}_w $(0 \leq j \leq n-1)$ is assigned the following operation:

If $(t_o, \ldots, t_{n-1}) \in W_\Sigma(X_v)^w$ and $c_{\langle w,v,w_j \rangle} \in D_{()}(\Omega)_{\langle\langle w,w_j\rangle\langle v,w_o\rangle \ldots \langle v,w_{n-1}\rangle, \langle v,w_j\rangle\rangle}$, then $c_{\langle w,v,w_j\rangle}(\delta^{j+1}_w, t_o, \ldots, t_{n-1}) = t_j$. That is, δ^{j+1}_w 'chooses' the (j+1)st element in the list $t_o, \ldots, t_{n-1}$.

<u>Example 3</u>: Let I = {1} and $\Omega_{\langle \lambda,1\rangle} = \{a, b, c\}$, $\Omega_{\langle 11,1\rangle} = \{f\}$, $\Omega_{\langle 11\ 1,1\rangle} = \{g\}$. We note that this example illustrates the alphabet for the usual notion of algebra. Since any $h \in \Omega$ is of type $\langle w,1 \rangle$, with

$w = 1 \ldots 1$, we can indicate this by writing $h \in \Omega_n$ for $\ell(w) = n$. This is more like the notation we encounter in the usual theory. Thus, in Ω above: $\Omega_{\langle \lambda,1\rangle} = \Omega_0$, $\Omega_{\langle 11,1\rangle} = \Omega_2$, $\Omega_{\langle 111,1\rangle} = \Omega_3$.

The derived algebra $D_{()}(W_\Omega)$ of the algebra W_Ω is obtained as follows:

(i) The sorting set is $D_{()}(I) = \{0, 2, 3\}$. (We have again used $n = \ell(w)$ instead of $\langle w,i\rangle$);

(ii) The indexed set of operator symbols is $D_{()}(\Omega)_{\langle \lambda, 0\rangle} = \{a,b,c\}$, $D_{()}(\Omega)_{\langle \lambda, 2\rangle} = \{f, \delta^1_2, \delta^2_2\}$, $D_{()}(\Omega)_{\langle \lambda, 3\rangle} = \{g, \delta^1_3, \delta^2_3, \delta^3_3\}$, and $c \in D_{()}(\Omega)_{\langle m\ n\ \ldots\ n,\ n\rangle}$, usually written $c_{\langle mn\rangle}$, for $m = 2, 3$ and $n = 0, 2, 3$. (We have of course used n instead of $w \in I^*$ such that $\ell(w) = n$.);

(iii) Let $X = \{x,y,z\}$ (we only need one sort of generator). Then the element of the carrier of $D_{()}(W_\Omega)$ of sort $\langle \lambda, 0\rangle$ is the set W_Ω, the element of sort $\langle \lambda, 2\rangle$ is the set $W_\Omega(\{x,y\})$, and the element of sort $\langle \lambda, 3\rangle$ is the set $W_\Omega(\{x,y,z\})$;

(iv) a, b, c, f, g name the constants a, b, c, fxy, gxyz respectively. δ^j_n and $c_{\langle mn\rangle}$ are assigned the obvious projection and composition operations.

(I am greatly indebted to J.W.Thatcher for parts of this section.)

Remark

Although it has not been explicitly stated until now, we are not assuming that each symbol in a many-sorted alphabet has a unique type. That is, the disjoint union of Σ may not be the same as the union of Σ. This corresponds, in Thatcher (1) for example, to allowing multiple ranking of symbols in the usual theory. This may be somewhat confusing at first and the reader may be reassured by the fact that, for his convenience, he may assume that, in fact, each symbol has a unique type. This assumption will not make a difference in most of the following. This is because we can associate an alphabet Σ' (with each symbol having a unique type) with Σ by assocating with each $f \in \Sigma$ the set $\{f_{\langle w,i \rangle} \in \Sigma'_{\langle w,i \rangle} | f \in \Sigma_{\langle w,i \rangle}\}$. Thus with each type $\langle w,i \rangle$ associated with the symbol $f \in \Sigma$, we have the element $f_{\langle w,i \rangle} \in \Sigma'$ with unique type $\langle w,i \rangle$ in the alphabet Σ'. It is easy to see how we may regard a Σ-algebra as a Σ'-algebra. Where the assumption does make a difference, we will state so explicitly, indicating the alternate version necessary.

II FORMAL LANGUAGES

II.0 Introduction

In this chapter we generalise some of the concepts of conventional formal language theory to the many-sorted case. The method of presentation is similar to what one finds in the usual exposition of the theory; although it is hoped that some of the material presented here clarifies the conventional theory. (References: Gross and Lentin, Hopcroft and Ullman, or any other basic book on formal language theory.)

II.1 Term Grammars

Let Σ be a finite many-sorted alphabet, sorted by I. Let $X = \{X_i\}_{i \varepsilon I}$ be any family of sets indexed by I. If $t \ \varepsilon \ \hat{W}_\Sigma(X)$, we define the depth of the term t, d(t) as follows:

(o) If $t \ \varepsilon \ X_i$, some $i \ \varepsilon \ I$, then $d(t) = 0$;

(i) If $t = ft_o \ldots t_{n-1}$, for $f \ \varepsilon \ \Sigma_{\langle w,i \rangle}$ and $t_j \ \varepsilon \ W_\Sigma(X)_{w_j}$ $(0 \leq j \leq n-1)$, then $d(t) = 1 + \max\limits_{0 \leq j \leq n-1} \{d(t_j)\}$.

Suppose $t_1 \ \varepsilon \ W_\Sigma(X)_i$, $t_2 \ \varepsilon \ W_\Sigma(X)_j$ for some $i, j \ \varepsilon \ I$. Then t_1 is said to be a subterm of t_2, $t_1 \leq t_2$, if there exists $t'_2 \ \varepsilon \ W_\Sigma(X, Y_i)_j$ and $Sub_i(t_2'; t_1) = t_2$. (We have introduced the notation $W_\Sigma(X, X')$ to indicate that the generators are the indexed family of sets $\{X_i \cup X_i'\}_{i \varepsilon I}$.) This Y_i ($i \ \varepsilon \ I^*$ a string of length one) is of course the set of variables $\{x_{o,i}\}$ ($i \ \varepsilon \ I$) and is sorted by I as in Example 1, Section I.4. t_1 is said to be a proper subterm of t_2, $t_1 < t_2$, just in the case that $t_1 \leq t_2$ and $t_1 \neq t_2$.

Let $a \ \varepsilon \ \Sigma_{\langle \lambda,i \rangle}$ (that is, a is an individual symbol of sort i). Given $t \ \varepsilon \ \hat{W}_\Sigma(X)$ and $t' \ \varepsilon \ W_\Sigma(X)_i$, we define the operation $t \cdot_a t'$ of a-substitution as follows: $t \cdot_a t' = Sub_i(t''; t')$ where $t = Sub_i(t''; a)$,

$t'' \varepsilon \hat{W}_{\Sigma}(X, Y_i)$, and t" has no occurrences of a (that is, as a string of symbols, t" does not contain an occurrence of the symbol a). We emphasize that this operation is defined only for a and t' of the same sort.

Again let $a \varepsilon \Sigma_{\langle\lambda,i\rangle}$. Given $U \sqsubseteq \hat{W}_{\Sigma}(X)$ and $V \sqsubseteq W_{\Sigma}(X)_i$, we define the operation, $U \cdot_a V$ of <u>a-complex product</u> as follows: $U \cdot_a V =$ $\{t \mid t = Sub_w(t''; t_o', \ldots, t'_{n-1});\ Sub_w(t''; \underbrace{a, \ldots, a}_{n\ \text{times}}) \varepsilon U$ with exactly n-occurrences of a; $t'_j \varepsilon V$ for $0 \leq j \leq n-1$; and $t'' \varepsilon \hat{W}_{\Sigma}(X, Y_w)$, with w a string of i's of length n, in which a does not occur$\}$. Intuitively, we obtain the set $U \cdot_a V$ by substituting some (not necessarily the same) element of V for each occurrence of the symbol a in an element of U. It is obvious that $U \cdot_a \{a\} = U$ and we define the set $U \cdot_a \phi$ to be the subset of U with no occurrences of a. We again emphasize that this operation is defined only for a and V of the same sort.

Let $a \varepsilon \Sigma_{\langle\lambda,i\rangle}$. Given $U \sqsubseteq W_{\Sigma}(X)_i$, we define the operation U^{*a} of <u>a-Kleene closure</u> as follows: Let $V^o = \{a\}$ and $V^{m+1} = V^m \bigcup U \cdot_a V^m$. Then $U^{*a} = \bigcup_{n \varepsilon \underline{N}} V^n$. We emphasize that this operation is defined only for a and U of the same sort.

<u>Example 1</u>: Consider the alphabet Σ of Example I·3·5. If t = +a+ab and t' = b, then $t \cdot_a t'$ = +b+bb, while if $t' = \lambda$ then $t \cdot_a t'$ is undefined (as t' and a are not the same sort). We note that if $U \sqsubseteq (W_{\Sigma})_i$, then $U \cdot_a \{a\} = U \cdot_b \{b\} = U \cdot_{\lambda} \{\lambda\} = U$.

Let $U = \{*a\lambda\} \sqsubseteq (W_{\Sigma})_o$. So $U^{*\lambda}$ is defined and we can calculate it as follows:

$$V^o = \{\lambda\};$$

$$\begin{aligned} V^1 &= V^o \bigcup U \cdot_{\lambda} V^o \\ &= \{\lambda\} \bigcup \{*a\lambda\} \cdot_{\lambda} \{\lambda\} \\ &= \{\lambda, *a\lambda\}; \end{aligned}$$

$$V^2 = V^1 \bigsqcup U \cdot_\lambda V^1$$
$$= \{\lambda, *a\lambda\} \bigsqcup \{*a\lambda\} \cdot_\lambda \{\lambda , *a\lambda\}$$
$$= \{\lambda, *a\lambda, *a*a\lambda\};$$
$$V^3 = V^2 \bigsqcup U \cdot_\lambda V^2$$
$$= \{\lambda, +a\lambda, *a*a\lambda, *a*a*a\lambda\};$$

etc.

Obviously $U^{*\lambda} = \bigsqcup_{n \in N} V^n = \{\lambda, *a\lambda, *a*a\lambda, *a*a*a\lambda, \text{etc.}\}$. Similarly, if $U = \{+ab\} \subseteq (W_\Sigma)_1$, then $U^{*a} = \{a, +ab, +a+ab, \text{etc.}\}$ whereas $U^{*\lambda}$ is now undefined.

For further examples and clarification, in the generalised case, the reader is referred to Thatcher and Wright.

We now wish to introduce two operations on many-sorted alphabets and study their effect on sets of terms. The first operation may be called 'change of symbol', although we will most often refer to it as projection. Let Ω and Δ be two many-sorted alphabets sorted by I such that $\Omega_{\langle w,i \rangle} = \phi$ if and only if $\Delta_{\langle w,i \rangle} = \phi$. That is, Ω and Δ have symbols of the same types. Consider a family of functions $\{\pi_{\langle w,i \rangle} : \Omega_{\langle w,i \rangle} \to \Delta_{\langle w,i \rangle}\}$ (usually written $\pi : \Omega \to \Delta$). Then a homomorphism $\overline{\pi} : D_{()}(W_\Omega) \to D_{()}(W_\Delta)$ is induced in the natural way. We are assuming that there are assignments to the projections which are the identity in each case. $\overline{\pi}$ is called a projection and $\overline{\pi}^{-1}$ is called an inverse projection. (We have in fact been cheating in calling $\overline{\pi} : D_{()}(W_\Omega) \to D_{()}(W_\Delta)$ a homomorphism. In actual fact, we have here a function between algebras over two different alphabets: $D_{()}(\Omega)$ and $D_{()}(\Delta)$. If we consider a new alphabet Σ which is obtained from $D_{()}(\Omega)$ ($D_{()}(\Delta)$) by the deletion of nullaries corresponding to $\Omega(\Delta)$ and regard the families Ω and Δ as generators, then in fact we do have a homomorphism $\phi: W_\Sigma(\Omega) \rightarrow W_\Sigma(\Delta)$. The function $\overline{\pi}: D_{()}(W_\Omega) \rightarrow D_{()}(W_\Delta)$ is a simple example of what Birkhoff calls a cryptomorphism.)

Let Ω and Δ be two many-sorted alphabets sorted by I. Let $V_{\Omega,\Delta} = \{w \in I^* | \Omega_{\langle w,i \rangle} \neq \phi$ or $\Delta_{\langle w,i \rangle} \neq \phi$ for some $i \in I\}$. Consider the indexed family of assignments $\{\phi_{\langle w,i \rangle}: \Omega_{\langle w,i \rangle} \rightarrow D_{V_{\Omega,\Delta}}(W_\Delta)_{\langle \lambda, \langle w,i \rangle \rangle}\}$ (which extends as the identity to projections of sort $\langle w, i \rangle$). Note that each $f \in \Omega_{\langle w,i \rangle}$ is assigned a derived operation of W_Δ of type $\langle w, i \rangle$. A homomorphism $\overline{\phi}: D_{()}(W_\Omega) \rightarrow D_{V_{\Omega,\Delta}}(W_\Delta)$ is induced in the natural way (with a misuse of the name 'homomorphism' similar to that in the definition of projection). The function $\overline{\phi}: D_{()}(W_\Omega) \rightarrow D_{V_{\Omega,\Delta}}(W_\Delta)$ is called a homomorphism with respect to composition, or, more briefly, a homomorphism and $\overline{\phi}^{-1}$ is called an inverse homomorphism. (The motivation for this definition will be given in Chapter VI.)

A semi-Thue production on terms (over the many-sorted alphabet Σ) is a pair $p = (\mathrm{Sub}_i(z; s), \mathrm{Sub}_i(z; s'))$ where $s, s' \in W_\Sigma(X', X_w)_i$ for some $w \in I^*$, no element of X_w occurs in s' which does not occur in s, and z is a variable which takes as values elements of $\hat{W}_\Sigma(X', Y_i)$ (Y_i is of course the set $\{x_{o,i}\}$ indexed by the string (of length one) $i \in I^*$). We define the relation $\underset{p}{\Longrightarrow}: \hat{W}(X') \rightarrow \hat{W}(X')$ as follows, for $t, t' \in W_\Sigma(X')_j$: $(t, t') \in \underset{p}{\Longrightarrow}$ (more commonly written $t \underset{p}{\Longrightarrow} t'$) if and only if there exist $\overline{t} \in \hat{W}_\Sigma(X', Y_i)$, $(\overline{t}_o, \ldots, \overline{t}_{n-1}) \in W_\Sigma(X')^w$ such that

$t = \mathrm{Sub}_i(\overline{t};\ \mathrm{Sub}_w(s;\ \overline{t}_o, \ldots, \overline{t}_{n-1}))$ and $t' = \mathrm{Sub}_i(\overline{t};\ \mathrm{Sub}_w(s';\ \overline{t}_o, \ldots, \overline{t}_{n-1}))$.

t' is called an immediate consequence of the term t. t is said to directly derive t' via the production p.

It is obvious that we can drop the variable z in the definition of semi-Thue production without ambiguity. We will adopt this convention from now on and write productions in the form (s, s') or, more commonly, $s \to s'$.

Example 2: Let Σ be as in Example 1. Then $p = (*a\lambda, \lambda)$ or $*a\lambda \to \lambda$ is an example of a semi-Thue production. Thus, for example, $*a\lambda \underset{p}{\Longrightarrow} \lambda$; that is λ is an immediate consequence of $*a\lambda$. We also have $*b*a\lambda \underset{p}{\Longrightarrow} *b\lambda$ and $*+ba*+bb*a*a\lambda \underset{p}{\Longrightarrow} *+ba*+bb*a\lambda \underset{p}{\Longrightarrow} *+ba*+bb\lambda$. Note that $*a\lambda$ is not an immediate consequence of λ (for which the pair p would have to be reversed) but is said to directly derive λ via the production p.

Consider $p = (*x*a\lambda, *x*+ax\lambda)$ or $*x*a\lambda \to *x*+ax\lambda$ where X is a set of generators and $x \in X_1$. Then, for instance, $*b*a\lambda \underset{p}{\Longrightarrow} *b*+ab\lambda$ since $*b*a\lambda = Sub_1(*x*a\lambda; b)$ and $*b*+ab\lambda = Sub_1(*x*+ax\lambda; b)$. We note that $*b*+aa\lambda$ is not an immediate consequence of $*b*a\lambda$ since not the same term has been substituted for each occurrence of x in $*x*+ax\lambda$ (b being substituted for the first occurrence of x and a for the second). Also note that $*x*+ax\lambda$ is not an immediate consequence of $*x*a\lambda$ (over the many-sorted alphabet Σ) since x is not in Σ.

A semi-Thue grammar (or system) is a 4-tuple $\langle \Omega, \Sigma, P, S \rangle$ where

(i) Σ is a finite many-sorted alphabet (sorted by I);

(ii) $\Omega_{\langle w,i \rangle} \subseteq \Sigma_{\langle w,i \rangle}$ for all $i \in I$ (usually will be written as $\Omega \subseteq \Sigma$). Ω is called the terminal alphabet while $N = \Sigma - \Omega$ (that is, $N = \{\Sigma_{\langle w,i \rangle} - \Omega_{\langle w,i \rangle}\}$) is called the non-terminal alphabet;

(iii) P is a finite set of semi-Thue productions; and

(iv) $S \subseteq \hat{W}_\Sigma$ is a finite set of axioms.

Define the relation $\underset{P}{\Longrightarrow} : \hat{W}_\Sigma(X') \to \hat{W}_\Sigma(X')$ as follows, for $t, t' \in W_\Sigma(X')_i : t \underset{P}{\Longrightarrow} t'$ (instead of $(t, t') \in \underset{P}{\Longrightarrow}$) if and only if there is some $p \in P$ such that $t \underset{p}{\Longrightarrow} t'$. Let $\overset{*}{\underset{P}{\Longrightarrow}} : \hat{W}_\Sigma(X') \to \hat{W}_\Sigma(X')$ be the least relation such that $\overset{*}{\underset{P}{\Longrightarrow}}$ contains $\underset{P}{\Longrightarrow}$ and is reflexive and transitive; that is $t \overset{*}{\underset{P}{\Longrightarrow}} t'$ if and only if there exist $t_o, \ldots, t_n \in W_\Sigma(X')_i$,

$p_o, \ldots, p_{n-1} \in P$ such that $t = t_o \underset{p_o}{\Longrightarrow} t_1 \underset{p_1}{\Longrightarrow} \cdots \underset{p_{n-1}}{\Longrightarrow} t_n = t'$. The relation $\overset{*}{\underset{P}{\Longrightarrow}}: \hat{W}_\Sigma(X') \to \hat{W}_\Sigma(X')$, called derivation, is the reflexive, transitive closure of the relation $\underset{P}{\Longrightarrow}: \hat{W}_\Sigma(X') \to \hat{W}_\Sigma(X')$. If $t \overset{*}{\underset{P}{\Longrightarrow}} t'$ then t' is said to be a consequence of t and t is said to derive t'.

The set generated by the grammar $G = \langle \Omega, \Sigma, P, S \rangle$ is $L(G) = \{t' \in W_\Omega | t \overset{*}{\underset{P}{\Longrightarrow}} t' \text{ for some } t \in S\}$. It should be obvious that we can replace G by a grammar $G' = \langle \Omega, \Sigma', P', \{Z\} \rangle$, where $\Sigma'_{\langle w,j \rangle} = \Sigma_{\langle w,j \rangle}$ for $w \neq \lambda$ or $w = \lambda$, $j \notin I_S = \{i \in I | S \cap (W_\Sigma)_i \neq \phi\}$; $\Sigma'_{\langle \lambda,i \rangle} = \Sigma_{\langle \lambda,i \rangle} \cup \{Z\}$ for $i \in I_S$ and Z a symbol not in Σ; and $P' = P \cup \{Z \to t | t \in S\}$, so that $L(G) = L(G')$. We will use this fact from now on and abuse our notation somewhat to define grammars as a 4-tuple $\langle \Omega, \Sigma, P, Z \rangle$ where Z is the axiom. We will also use the symbols $\overset{*}{\Longrightarrow}$, $\Longrightarrow$ instead of $\overset{*}{\underset{P}{\Longrightarrow}}$, $\underset{P}{\Longrightarrow}$ respectively whenever it is obvious which set of productions P we have in mind.

Remark: Of course, if we assume that each symbol in an alphabet Σ has a unique type, then we cannot associate such a grammar G' with the grammar G. However, let $G'' = \langle \Omega, \Sigma'', P'', Z \rangle$ be the grammar where $\Sigma''_{\langle w,j \rangle} = \Sigma_{\langle w,j \rangle}$ for $w \neq \lambda$ or $w = \lambda$, $j \notin I_S = \{i \in I | S \cap (W_\Sigma)_i \neq \phi\}$; $\Sigma''_{\langle \lambda,i \rangle} = \Sigma_{\langle \lambda,i \rangle} \cup \{Z_i\}$ for $i \in I_S$; $Z = \{Z_i | i \in I_S \text{ and } Z_i \text{ not in } \Sigma\}$; and $P'' = P \bigcup_{i \in I_S} \{ Z_i \to t | t \in S \cap (W_\Sigma)_i\}$. Then $L(G) = L(G')$ and we have a unique axiom for each sort in S. The reader is asked to take account of this difference between G' and G'' as we proceed.

We obtain different classes of grammars (and sets generated by these grammars) by restricting the types of productions we allow in a grammar. A regular grammar $G = \langle \Omega, \Sigma, P, Z \rangle$ is one in which all $p \in P$ are of the form $A \to fA_o \ldots A_{n-1}$ ($f \in \Omega_{\langle w,i \rangle}$, $A \in N_{\langle \lambda,i \rangle}$, and

$A_j \in N_{\langle\lambda,w_j\rangle}$ for $0 \leq j \leq n-1$) or $A \to a(a \in \Omega_{\langle\lambda,i\rangle}$, $A \in N_{\langle\lambda,i\rangle}$) where $N_{\langle w,j\rangle} = \phi$ for $w \neq \lambda$. The former set of productions is called <u>non-terminal</u> while the latter set is called <u>terminal</u>. The sets generated by such grammars are said to be <u>regular</u> (see Brainerd (1), (2)).

A <u>context-free grammar</u> $G = \langle \Omega, \Sigma, P, Z \rangle$ is one in which all $p \in P$ are of the form $A(x_o, \ldots, x_{n-1}) \to t$ (or, more correctly, $Ax_o \ldots x_{n-1} \to t$) where $A \in N_{\langle w,i\rangle}$, $X_w = \{x_o, \ldots, x_{n-1}\}$, and $t \in W_\Sigma(X_w)_i$. (Note that we have reindexed X_w for simplicity of notation.) The sets generated by such grammars are said to be <u>context free</u> (see Rounds (1), (2)).

A <u>monotonic grammar</u> $G = \langle \Omega, \Sigma, P, Z \rangle$ is one in which all $p \in P$ are either of the form $t \to t'$, where $t, t' \in W_\Sigma(X_w)_i$, some $w \in I^*$, and $d(t) \leq d(t')$ (hence the name monotonic) or $Ax_{o,w_o} \ldots x_{n-1,w_{n-1}} \to x_{j,w_j}$ $(0 \leq j \leq n-1)$. The sets generated by such grammars are said to be <u>context sensitive</u>.

We note that the class of regular grammars is a subclass of the class of context free grammars which is in turn a subclass of the class of monotonic grammars.

We give below an example of a semi-Thue grammar which is not monotonic. For the sake of simplicity our example uses a non-sorted alphabet (as explained in Example I.3.4). Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $\Omega_o = \{v_o, v_1\}$, $\Omega_1 = \{\neg\}$, $\Omega_2 = \{\vee, \wedge\}$;

(ii) $N_o = \{Z, V\}$, $N_1 = \{C, F\}$; and

(iii) $P = P_0 \cup P_1$ where:

(a) $P_0 = \{Z \to F(\wedge(V, V)), Z \to F(\vee(V, V)), Z \to F(\neg(V)),$
$V \to \wedge(V, V), V \to \vee(V, V), V \to \neg(V), V \to v_o, V \to v_1\}$ and

(b) $P_1 = \{F(\wedge(x_o, x_1)) \to \wedge(F(x_o), F(x_1)),$
$F(\vee(x_o, x_1)) \to \vee(F(x_o), F(x_1)), F(\neg(x_o)) \to C(x_o), F(v_o) \to v_o,$
$F(v_1) \to v_1, C(v_o) \to v_o, C(v_1) \to v_1, C(\neg(x_o)) \to F(x_o),$
$C(\vee(x_o, x_1)) \to \wedge(C(x_o), C(x_1)), C(\wedge(x_o, x_1)) \to \vee(C(x_o), C(x_1))\}.$

This grammar generates all the Boolean expressions, in the Boolean variables v_o and v_1, such that the only subexpressions which are complemented are the variables. The productions in P_o are used to generate all Boolean expressions in the Boolean variables v_o and v_1, while those in P_1 convert such expressions, using de Morgan's laws, into equivalent ones so that the variables are the only subexpressions occurring with complement signs on them. This grammar is not monotonic since there are productions $t_o \to t_1$ in P such that $d(t_o) > d(t_1)$ $(F(\neg(x_o)) \to C(x_o)$, for example).

Note that we have adopted the common practice of using commas and brackets in writing our Σ-expressions. This will be done in most of what follows for the sake of clarity.

Examples of regular and context free grammars can be found in Chapters III and IV, respectively.

Remark: Let Σ be a finite set (and Σ^* the set of strings over Σ). Let $T(\Sigma)$ be the family of ranked symbols with $T(\Sigma)_0 = \{\lambda\}$, $T(\Sigma)_1 = \Sigma$ and $T(\Sigma)_n = \phi$ for $n > 1$. There exists a bijective function $\phi_\Sigma : \Sigma^* \to W_{T(\Sigma)}$ (and we will often write ϕ_Σ instead of ϕ_Σ^{-1} because of this bijection property) given by the inductive definition:

(0) $\phi_\Sigma(\lambda) = \lambda$;

(i) $\phi_\Sigma(w) = w\lambda$ for $w \in \Sigma^+$.

We call the elements of the carrier of $W_{T(\Sigma)}$ monadic or unary terms.

Thus, given a regular grammar $G = \langle \Omega, \Sigma, P, Z \rangle$ (of conventional theory), we can easily convert it into a regular term grammar

$G' = \langle T(\Omega), T(\Sigma), P', Z \rangle$. This is accomplished as follows:

(i) $T(\Sigma)_0 - T(\Omega)_0 = \Sigma - \Omega$ and $T(\Sigma)_1 - T(\Omega)_1 = \phi$;

(ii) P' is obtained from P by

(a) replacing $A \to a$ in P by $A \to a\lambda$ in P';

(b) replacing $A \to aB$ in P by $A \to aB$ in P';

and (c) replacing $Z \to \lambda$ in P by $Z \to \lambda$ in P';

(iii) Replace each production of the form $A \to a\lambda$ in P' by two regular productions with the same effect (for example, $A \to aL$, $L \to \lambda$).

It should be clear that $\phi_\Sigma(L(G)) = L(G')$.

If $G = \langle \Omega, \Sigma, P, Z \rangle$ is a context free grammar (of conventional theory) in Chomsky normal form, we obtain a context free term grammar $G' = \langle T(\Omega), T(\Sigma), P', Z \rangle$ as follows:

(i) Let $T(\Sigma)_0 - T(\Omega)_0 = \{Z\}$ and $T(\Sigma)_1 - T(\Omega)_1 = \Sigma - (\Omega \cup \{Z\})$;

(ii) P' is obtained from P by

(a) replacing $Z \to AB$ in P by $Z \to A(B(\lambda))$ in P';

(b) replacing $Z \to a$ in P by $Z \to a(\lambda)$ in P';

(c) replacing $Z \to \lambda$ in P by $Z \to \lambda$ in P';

(d) replacing $A \to BC$, $A \neq Z$, in P by $A(x) \to A(B(x))$ in P';

and (e) replacing $A \to a$, $A \neq Z$, in P by $A(x) \to a(x)$ in P'.

Again it should be clear that $\phi_\Sigma(L(G)) = L(G')$.

If $G = \langle \Omega, \Sigma, P, Z \rangle$ is a monotonic grammar (of conventional theory), we obtain a monotonic term grammar $G' = \langle T(\Omega), T(\Sigma), P', Z \rangle$ as follows:

(i) Let $T(\Sigma)_0 - T(\Omega)_0 = \{Z\}$ and $T(\Sigma)_1 - T(\Omega)_1 = \Sigma - (\Omega \cup \{Z\})$;

(ii) P' is obtained from P by:

(a) replacing $Z \to t$ in P $(t \neq \lambda)$ by $Z \to t\lambda$ in P';

(b) replacing $Z \to \lambda$ in P by $Z \to \lambda$ in P';

and (c) replacing $A \to t(A \neq Z)$ in P by $A(x) \to t(x)$ in P'.

Again it should be clear that $\phi_\Sigma(L(G)) = L(G')$. (We have assumed in all the above that the empty string λ appears only in productions of the form $Z \to \lambda$, Z the axiom.)

Example 4: We illustrate the above procedures with the following simple example. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $\Omega = \{a, b\}$ and $\Sigma - \Omega = \{Z, A, B, C, D\}$;

(ii) $P = \{Z \to AB, Z \to AC, C \to DB, D \to AC, D \to AB, A \to a, B \to b\}$.

G is a context free (string) grammar and $L(G) = \{a^n b^n | n > 0\}$.

We apply the above procedure to get the grammar $G' = \langle T(\Omega), T(\Sigma), P', Z \rangle$ where $P' = \{Z \to A(B(\lambda)), Z \to A(C(\lambda))$, $C(x) \to D(B(x)), D(x) \to A(C(x)), D(x) \to A(B(x)), A(x) \to a(x), B(x) \to b(x)\}$. $L(G') = \{a^n b^n \lambda | n > 0\}$ and so $L(G') = \phi_\Sigma(L(G))$.

Note that what we have called the conventional definition of monotonic grammars is usually given as the definition of context sensitive grammars. The latter, however, is a different class of grammar whose productions are of the form $uAw \to uvw$ (to be read as: Replace A, in the context of u and w, by v) where $A \in \Sigma - \Omega$ and $u, v, w \in \Sigma^*$. It can be shown, however, that the two classes of grammars generate the same class of sets (of strings). Motivated by the above, we define a context sensitive term grammar $G = \langle \Omega, \Sigma, P, Z \rangle$ ($\Sigma \supseteq \Omega$ a many-sorted alphabet) to be one in which all productions $p \in P$ are of the form $t \to t'$ where $t, t' \in W_\Sigma(X)_i$ and there exist $A \in N_{\langle w,j \rangle}$, $r \in W_\Sigma(X_w)_j$, $s \in W_\Sigma(X, Y_j)_i$, and $(s_o, \ldots, s_{n-1}) \in W_\Sigma(X)^w$ such that $t = Sub_j(s; As_o \ldots s_{n-1})$ and $t' = Sub_j(s; Sub_w(r; s_o, \ldots, s_{n-1}))$. Although we will not prove it here, it can be shown that context sensitive and monotonic term grammars generate the same class of sets (of terms).

III. RECOGNIZABLE/REGULAR/EQUATIONAL SETS

III.0 Introduction

In this chapter we present familiar material about recognizable sets (Thatcher and Wright, Doner), regular grammars (Brainerd (1)), regular sets (Thatcher and Wright), and equational sets (Mezei and Wright, Eilenberg and Wright) in a new setting: the many-sorted case. (The references are to studies of the generalised case. The reader is probably familiar with a great deal of material on the conventional case.) Since the material is not really new (and every effort is made by means of our notation to show this), we do not give full proofs of our results unless the proofs are new or are an improvement on the ones to be found in the literature.

We conclude the chapter with a very important theorem relating the classes of sets recognized by Σ-automata, generated by regular grammars over Σ, regular over Σ, and equational over Σ.

III.1 Finite Automata

Let Σ be a many-sorted alphabet, sorted by I. We wish to define finite automata which 'accept' Σ-expressions and motivated by this we begin by defining Σ to be the possible set of input symbols and $\hat{W}_\Sigma$ to be the universe of possible input terms.

A (deterministic) finite automaton of type Σ or, more briefly, a Σ-automaton, is a finite Σ-algebra. If A is such a Σ-automaton, then $\hat{A}$ is the set of states; for $a \in \Sigma_{\langle \lambda,i \rangle}$, a_A is an initial state; and for $f \in \Sigma_{\langle w,i \rangle}$, f_A is the (direct) transition function for the input symbol f.

A non-deterministic finite automaton of type Σ, or, more briefly, a non-deterministic Σ-automaton, is a finite Σ-structure. Given such a structure M, a_M is the set of initial states corresponding to the input symbol $a \in \Sigma_{\langle \lambda,i \rangle}$ and f_M is the transition relation for the input symbol $f \in \Sigma_{\langle w,i \rangle}$.

Each input term $t \in (W_\Sigma)_i$ induces a corresponding output state $\eta_i^A(t)$ in a Σ-automaton A (where $\eta^A : W_\Sigma \to A$ is the unique homomorphism defined by The Fundamental Theorem of Algebra). Similarly, each input term $t \in (W_\Sigma)_i$ induces a corresponding set of output states $\eta_i^{pM}(t)$ in a non-deterministic Σ-automaton M (where $\eta_i^{pM} : W_\Sigma \to pM$ is the unique homomorphism).

As is the case in both the conventional and the generalised theories of finite automata, we can define the set of terms recognised by an automaton as follows:

(i) For any deterministic Σ-automaton A and a choice of final states $\{A_i^F \sqsubseteq A_i\}$ (usually written $A^F \sqsubseteq A$), the behaviour of A with respect to A^F is the indexed family of sets $bh_A(A^F) = \{t \in (W_\Sigma)_i \mid \eta_i{}^A(t) \in A_i^F\}$;

(ii) For any non-deterministic Σ-automaton M and a choice of final states $\{M_i^F \sqsubseteq M_i\}$ (usually written $M^F \sqsubseteq M$), the behaviour of M with respect to M^F is the indexed family of sets $bh_M(M^F) = \{t \in (W_\Sigma)_i \mid \eta_i^{pM}(t) \sqcap M_i^F \neq \phi\}$;

(iii) An indexed family of sets $U \sqsubseteq W_\Sigma$ is recognizable if there exists a Σ-automaton A (deterministic or non-deterministic) and a choice of final states $A^F \sqsubseteq A$ such that $bh_A(A^F) = U$.

Example 1: (a) Let Σ be the many-sorted alphabet of Example I.3.5. Let M be a Σ-automaton such that -

(i) $M_o = \{Z, E\}$, $M_1 = \{A, B, C, D, F, G\}$;

(ii) (A, D, C), (C, B, D), (A, B, C), (A, B, F), (A, D, G) $\in +_M$ and (F, E, Z), (G, E, Z) $\in *_M$;

(iii) The set of initial states corresponding to $a \in \Sigma_{\langle\lambda,1\rangle}$ is the set {A}, to $b \in \Sigma_{\langle\lambda,1\rangle}$ is the set {B}, and to $\lambda \in \Sigma_{\langle\lambda,0\rangle}$ is the set {Z, E};

(iv) The choice of final states is $\{Z\} \subseteq M_o$ and $\phi \subseteq M_1$.

M is obviously a non-deterministic automaton and it recognizes the indexed family of sets $\{\{\lambda, *{+}ab\lambda, *{+}a{+}{+}abb\lambda, *{+}a{+}{+}a{+}{+}abbb\lambda, \text{etc.}\}, \phi\}$.

(b) If we changed the above example by allowing $Z \in M_1$ and $(A, B, Z), (A, D, Z) \in +_M$ with the choice of final states $\{Z\} \subseteq M_o$, $\{Z\} \subseteq M_1$, then M recognizes the indexed family of sets $\{\{\lambda, *{+}ab\lambda, *{+}a{+}{+}abb\lambda, \text{etc.}\}, \{+ab, +a{+}{+}abb, +a{+}{+}a{+}{+}abbb, \text{etc.}\}\}$.

We are now ready to state three important theorems whose proofs, since they are of a purely algebraic nature, are exactly analogous to those found in the conventional and generalised theories. As such, we restrict our 'proofs' to giving brief versions of the necessary constructions used and leave it to the reader to supply his own detailed proofs.

<u>Theorem 1</u>. An indexed family of sets $U \subseteq W_\Sigma$ is recognizable by a deterministic Σ-automaton if and only if U is recognizable by a non-deterministic Σ-automaton.

<u>Proof</u>. If $U \subseteq W_\Sigma$ is recognizable by a deterministic Σ-automaton, then it is trivially recognizable by a non-deterministic Σ-automaton since Σ-algebras are Σ-structures (see Section I.3).

Conversely, if U is recognizable by some non-deterministic Σ-automaton M, then U is also recognizable by the deterministic Σ-automaton pM, the raised algebra of M.

From now on, we may refer to recognizable sets without reference to the kind of automaton involved.

Theorem 2: $U \sqsubseteq W_\Sigma$ is recognizable if and only if U is the union of classes of a finite congruence on W_Σ. (A congruence q on W_Σ is said to be finite if each q_i has a finite number of q_i-classes (and I is finite, as it must be here).) Note: We have here extended the notion of union from sets to indexed families of sets. For example, if U, V are two families of sets indexed by I, then $U \sqcup V = \{U_i \cup V_i\}_{i\epsilon I}$. Intersection, set difference (or complementation) and other operations on sets, can be extended in a similar way.

Proof. Let A be a (deterministic) Σ-automaton which recognizes U (with the choice of final states $A^F \sqsubseteq A$). We know from Chapter I that $\eta^A : W_\Sigma \to A$ induces a congruence on W_Σ. The classes of this congruence for each $i \epsilon I$ are $(\eta_i^A)^{-1}$ (a) for all $a \epsilon A_i$. Then

$U = \{\bigcup_{a\epsilon A_i^F} (\eta_i^A)^{-1}(a)\}_{i\epsilon I}$.

Conversely, let q be a finite congruence on W_Σ and let

$U = \{\bigcup_{k=0}^{m_i-1} x_k{}^q | m_i \geq 0\}_{i\epsilon I}$. Then $A = W_\Sigma/q$ is a finite automaton and if we let $A^F = \{\{x_o{}^q, \ldots, x^q_{m_i-1}\}\}_{i\epsilon I}$ then $bh_A(A^F) = U$.

Theorem 3. If $U, V \sqsubseteq W_\Sigma$ are recognizable, then so are $U \sqcap V$ and $W_\Sigma - U$.

Proof. If A, B are Σ-automata such that $bh_A(A^F) = U$, $bh_B(B^F) = V$ respectively then (i) $bh_A(A - A^F) = W_\Sigma - U$

and (ii) $bh_{A\times B}(A^F \times B^F) = U \sqcap V$.

Corollary. If $U, V \sqsubseteq W_\Sigma$ are recognizable, then so is $U \sqcup V$.

The last theorem proves the closure of the recognizable sets under the (extended) Boolean operations.

An automaton A, with choice of final states A^F, is said to be <u>equivalent</u> to an automaton B, with choice of final states B^F, if $bh_A(A^F) = bh_B(B^F)$.

One of our basic tasks in automata theory is to prove the existence or non-existence of effective procedures to solve certain questions about the sets accepted by certain classes of automata. These so-called decidability results will be the subject matter of the remainder of this section. We will be discussing three particular questions (given Σ-automata A and B and choices of final states A^F and B^F, respectively):

(i) Does $bh_A(A^F) = \phi$? (ϕ is of course $\{\phi\}_{i \varepsilon I}$);

(ii) Is $bh_A(A^F)$ infinite? (Here, by 'infinite' we mean: Is some component of $bh_A(A^F)$ infinite?)

(iii) Is $bh_A(A^F) = bh_B(B^F)$?

We proceed by first proving a number of auxiliary results which are of importance in themselves.

It is a well known theorem of conventional finite automata theory that, if in accepting a string $w_o w_1 w_2$ (over some given alphabet), an automaton is in the same state immediately after processing both strings w_o and $w_o w_1$, then the automaton will recognize $w_o w_1{}^k w_2$ for any natural number k. A generalisation of this to the many-sorted case is:

<u>Theorem 4</u>: Let A be a Σ-automaton and $t_o, t_1 \varepsilon (W_\Sigma)_i$ such that $\eta_i^A(t_o) = \eta_i^A(t_1)$. Given $\overline{t} \varepsilon W_\Sigma(X_i)_j$ such that $Sub_i(\overline{t}; t_o) = t$ and $Sub_i(\overline{t}; t_1) = t'$, then $\eta_j^A(t) = \eta_j^A(t')$. (That is, if t' is obtained from t by 'replacing an occurrence' of t_o in t by t_1 and if t_o and t_1 induce the same output state in the automaton A, then so will t and t'.)

Proof: Suppose $t = a \in \Sigma_{\langle\lambda,j\rangle}$. Then $t_o = t_1 = t = t' = a$ and the result follows. If $t \neq a \in \Sigma_{\langle\lambda, j\rangle}$, then $\eta_j^A(t) = \eta_j^A(t') = \overline{\phi}(\overline{t})$ where $\overline{\phi}: W_\Sigma(X_i) \to A$ is the homomorphism generated by the assignment $\phi_i: x_{o,i} \to \eta_i^A(t_o)$ $(= \eta_i^A(t_1))$ and again the result follows.

Corollary. Let A be a Σ-automaton and $(t_o, \ldots, t_{n-1}) \in (W_\Sigma)^w$, $(s_o, \ldots, s_{n-1}) \in (W_\Sigma)^w$ such that $\eta_{w_i}^A(t_i) = \eta_{w_i}^A(s_i)$ for $0 \leq i \leq n-1$. Given $\overline{t} \in W_\Sigma(X_w)_j$ such that $Sub_w(\overline{t};\ t_o, \ldots, t_{n-1}) = t$ and $Sub_w(\overline{t};\ s_o, \ldots, s_{n-1}) = t'$, then $\eta_j^A(t) = \eta_j^A(t')$.

Theorem 5. Suppose A is a Σ-automaton and that A has n states. (That is, the cardinality of $\hat{A}$ is n). Given $t \in (W_\Sigma)_i$, we can find $t' \in (W_\Sigma)_i$ such that $d(t') \leq n$ and $\eta_i^A(t) = \eta_i^A(t')$.

Proof. If $d(t) \leq n$, then we are done. Suppose, therefore, that $d(t) > n$. Then there exist $t_o, \ldots, t_{d(t)}$ such that $t = t_o > \ldots > t_{d(t)} = a \in \Sigma_{\langle\lambda,j\rangle}$. The corresponding sequence of states $\eta_{i_o}^A(t_o), \ldots, \eta_{i_{d(t)}}^A(t_{d(t)})$ must contain a repetition, say $\eta_{i_j}^A(t_j) = \eta_{i_k}^A(t_k)$ for $j<k$. Then, by the above theorem, if we replace t_{i_j} by t_{i_k} in t to obtain t', $\eta_{i_o}^A(t) = \eta_{i_o}^A(t')$. Now $d(t') \leq \ell(t') < \ell(t)$ (where $\ell(t)$ is the length of the string of symbols t). Thus, if $d(t') > n$, a finite number of repetitions (limited by $\ell(t')$) of the above procedure will result in a term of the required depth.

We are now ready to state our promised result:

Theorem 6. Given Σ-automata A and B and choices of final states A^F and B^F, respectively, there exist effective procedures for deciding the following questions:

(i) Does $bh_A(A^F) = \phi$?

(ii) Is $bh_A(A^F)$ infinite?

(iii) Are A and B equivalent?

Proof.

(i) Consider all terms $t \in \hat{W}_\Sigma$ such that $d(t) \leq n$, where n is the number of states of A. There are only a finite number of such terms. By the previous theorem, if $bh(A^F)_i$ contains any terms, then it contains terms of depth $\leq n$. Thus the procedure is to test all terms t such that $d(t) \leq n$ and we have $bh_A(A^F) \neq \phi$ if and only if $\eta_i^A(t) \in A_i^F$ for some such t and some $i \in I$.

(ii) If we show that $bh_A(A^F)$ is infinite if and only if A recognises a term t such that $n < d(t) \leq 2n$, then the obvious procedure is to test the finite number of terms of depth between n+1 and 2n for recognition by A. It remains to prove the condition of the above statement.

Suppose $t \in bh_A(A^F)_j$ and $n < d(t) \leq 2n$. Then, by the previous theorem, there exist $t_o, t_1 \in (W_\Sigma)_i$ such that $t_o < t_1$ and $\eta_i^A(t_o) = \eta_i^A(t_1)$. Let $\bar{t} \in W_\Sigma(X_i)_j$ such that $t = Sub_i(\bar{t}; t_o)$. It follows that $\eta_j^A(t) = \eta_j^A(Sub_i(\bar{t}; t_o)) = \eta_j^A(Sub_i(\bar{t}; t_1))$ and so $t' = Sub_i(\bar{t}; t_1) \in bh_A(A^F)_j$. Let $\bar{\bar{t}} \in W_\Sigma(X_i)_j$ such that $Sub_i(\bar{\bar{t}}; t_o) = t'$. It follows that $\eta_j^A(t) = \eta_j^A(t') = \eta_j^A(t'')$ (where $t'' = Sub_i(\bar{\bar{t}}; t_1)$) so $t'' \in bh_A(A^F)_j$. And so on. Thus $bh_A(A^F)$ is infinite.

Conversely, suppose $bh_A(A^F)$ is infinite and that no term of depth between n+1 and 2n is recognized by A. Let t be such that $d(t) > 2n$ and such that $d(t) = \min_{t \in \hat{W}_\Sigma}\{d(t) | t \in bh_A(A^F)_i$, some $i \in I$, and $d(t) > 2n\}$. (Such a term must exist if $bh_A(A^F)$ is infinite.) Then we can obtain terms $t_o, t_1 \in (W_\Sigma)_i$ such that $t > t_1 > t_o$ and $\eta_i^A(t_o) = \eta_i^A(t_1)$. Let $\bar{t} \in \hat{W}_\Sigma(X_i)$ such that $Sub_i(\bar{t}; t_1) = t$. Then A recognizes $t' = Sub_i(\bar{t}; t_o)$. t_o and t_1 can be so chosen that $d(t_o) \leq n$. This leads to a contradiction:

either to $n < d(t') \leq 2n$ or to $2n < d(t') < d(t\ )$. Thus there is a t, $n < d(t) \leq 2n$, recognized by A.

(iii) $(bh_A(A^F) \cap (W_\Sigma - bh_B(B^F))) \cup ((W_\Sigma - bh_A(A^F)) \cap bh_B(B^F))$ is $bh_C(C^F)$ for some Σ-automaton C and choice of final states C^F.

It can be seen that $bh_C(C^F) \neq \phi$ if and only if $bh_A(A^F) \neq bh_B(B^F)$ and so we can use (i) to decide the question.

III.2 Regular Grammars

We begin this section with an example of a regular grammar:

Example 1: Let $G = \langle \Sigma, \Delta, P, Z \rangle$ where:

(i) Σ is the alphabet of Example I.3.5;

(ii) $N_{\langle \lambda, 0 \rangle} = \{Z, E\}$, $N_{\langle \lambda, 1 \rangle} = \{A, B, C, D, F, G\}$;

(iii) $P = \{Z \to \lambda,\ Z \to *FE,\ Z \to *GE,\ C \to +AD,\ C \to +AB,\ D \to +CB,\ E \to \lambda,\ F \to +AB,\ G \to +AD,\ A \to a,\ B \to b\}$.

The set generated by G is $L(G) = \{\lambda, *{+}ab\lambda, *{+}a{+}{+}abb\lambda, *{+}a{+}{+}a{+}{+}abbb\lambda$, etc.$\}$.

If we modify the above example by allowing $Z \in N_{\langle \lambda, 1 \rangle}$ and adding the productions $Z \to +AB$ and $Z \to +AD$ to P, then $L(G) = \{\{\lambda, *{+}ab, *{+}a{+}{+}abb\lambda$, etc.$\}, \{+ab, +a{+}{+}abb, +a{+}{+}a{+}{+}abbb$, etc.$\}\}$. (L(G) is now an indexed family of sets.)

We proceed with the following classical result of finite automata theory (Chomsky and Miller, Brainerd (1)):

Theorem 1. The class of (indexed families of) sets recognized by Ω-automata is the same as the class of (indexed families of) sets generated by regular grammars over the alphabet Ω.

Proof. The theorem is an immediate consequence of the following lemmas:

Lemma 2. Given a regular grammar $G = \langle \Omega, \Sigma, P, Z \rangle$, one can effectively find an Ω-automaton A and a choice of final states $A^F \sqsubseteq A$ such that $L(G) = bh_A(A^F)$.

Proof. Given G, let the (non-deterministic) Ω-automaton be $A = \{X' | X \in N_{\langle \lambda, i \rangle}\}_{i \in I}$. The relations of A are $(Y'_o, \ldots, Y'_{n-1}, X') \in f_A$ for each $X \to fY_o \ldots Y_{n-1}$ in P (where $f \in \Omega_{\langle w,i \rangle}$, $X \in N_{\langle \lambda, i \rangle}$, and $(Y_o, \ldots, Y_{n-1}) \in N^w$). The choice of final states for A is $\{Z'\}$. (Note that $\{Z'\}$ is an indexed family of sets with each element of the family corresponding to each type of Z in N.)

It remains to show that, given $t \in (W_\Omega)_i$, $Z \overset{*}{\Longrightarrow} t$ if and only if $\eta_i^A(t) \sqcap \{Z'\} \neq \phi$ (where $Z' \in A_i$). We will prove the more general result: $X \overset{*}{\Longrightarrow} t$ if and only if $\eta_j^A(t) \sqcap \{X'\} \neq \phi$ for $t \in (W_\Omega)_j$.

If $t = a \in \Omega_{\langle \lambda, j \rangle}$ then we have

$X \overset{*}{\Longrightarrow} a$ if and only if $X \Longrightarrow a$ in G

if and only of $X \to a$ is in P

if and only if $\{X'\} \in a_A$

if and only if $\eta_j^A(a) \sqcap \{X'\} \neq \phi$.

If $t \neq a \in \Omega_{\langle \lambda, j \rangle}$, then $t = ft_o \ldots t_{n-1}$ for $f \in \Omega_{\langle w, j \rangle}$, $(t_o, \ldots, t_{n-1}) \in (W_\Omega)^w$ and we have

$X \overset{*}{\Longrightarrow} t$ if and only if there exists $(X_o, \ldots, X_{n-1}) \in N^w$ and $X \Longrightarrow fX_o \ldots X_{n-1} \overset{*}{\Longrightarrow} ft_o \ldots t_{n-1}$

if and only if $X \to fX_o \ldots X_{n-1}$ is in P and $X_k \overset{*}{\Longrightarrow} t_k$ for $0 \leq k \leq n-1$

if and only if $(X_o, \ldots, X_{n-1}, X) \in f_A$ and $X'_k \in \eta_{w_k}^{pA}(t_k)$ for $0 \leq k \leq n-1$

if and only if $\{X'\} \sqcap \{\eta_j^{pA}(t)\} \neq \phi$.

Proof. Let $G^1 = \langle \Omega, \Sigma^1, P^1, Z^1 \rangle$, $G^2 = \langle \Omega, \Sigma^2, P^2, Z^2 \rangle$ be grammars for U, V respectively. Assume, without loss of generality, that $N^1 \bigcap N^2 = \phi$. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be defined as follows:

(i) $N = N^1 \bigcup N^2$;

(ii) Let $Q = \{p \in P^1 | p = A \to a \text{ for some } A \in N^1_{\langle \lambda, i \rangle}\}$. Then $P = P^2 \bigcup (P^1 - Q) \bigcup \{A \to t | A \to a \in Q \text{ and } Z^2 \to t \in P^2\}$;

(iii) $Z = Z^1$.

It is easy to see that G is a regular grammar. Also, we have $L(G) = U \cdot_a V$ as any term generated by G is just a term in $L(G^1)$ with each a replaced by some (not necessarily the same) term in $L(G^2)$.

Similarly, let $G' = \langle \Omega, \Sigma', P', Z' \rangle$ be defined as follows:

(i) $N' = N^1$;

(ii) Let Q be as above. Then $P' = \{Z \to a\} \bigcup (P^1 - Q) \bigcup \{A \to t | A \to a \in Q \text{ and } Z^1 \to t \in P^1\}$;

(iii) $Z' = Z^1$.

Again G' is a regular grammar and $L(G') = U^{*a}$.

We now wish to study the operation of 'change of symbols' on recognizable (families of) sets. (Also see Thatcher and Wright.)

Theorem 5. If U, V are recognizable (families of) sets over the alphabets Ω and Δ, respectively, and $\overline{\pi}$ is a projection generated by the function $\pi : \Omega \to \Delta$, then $\{\overline{\pi}(U_i)\}_{i \in I}$ and $\{\overline{\pi}^{-1}(V)\}_{i \in I}$ (usually written as $\overline{\pi}(U)$ and $\overline{\pi}^{-1}(V)$, respectively) are recognizable (families of) sets over the alphabets Ω and Δ, respectively.

Proof. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a grammar for U. Define $G' = \langle \Delta, \Sigma', P', Z' \rangle$ as follows:

(i) $\Sigma' = N \bigcup \Delta$;

Conversely we have:

Lemma 3. Given an Ω-automaton A and a choice of final states $A^F \subseteq A$, one can effectively find a regular grammar $G = \langle \Omega, \Sigma, P, Z' \rangle$ such that $L(G) = bh_A(A^F)$.

Proof. Assume without loss of generality that A is deterministic and that each component of A^F is at most a singleton (say $Z \in A_i^F$ if and only if $A_i^F \neq \phi$). G is obtained as follows:

(i) Let $N = \{X' \in N_{\langle \lambda, i \rangle} | X \in A_i\}_{i \in I}$;

(ii) $P = \{Z' \to a | n_i^A(a) = Z, i \in I\} \cup \{Z' \to fX_o' \ldots X_{n-1}' | f \in \Omega_{\langle w,i \rangle}$ and $f_A(X_o, \ldots, X_{n-1}) = Z\} \cup \{X' \to fX_o' \ldots X_{n-1}' | f \in \Omega_{\langle w,i \rangle}$ and $f_A(X_o, \ldots, X_{n-1}) = X\}$

$= \{X' \to fX_o' \ldots X_{n-1}' | f \in \Omega_{\langle w,i \rangle}$ and $f_A(X_o, \ldots, X_{n-1}) = X\}$.

It remains to show that any term recognized by A has a derivation in G and conversely. This fact follows from the more general statement $n_i^A(t) = X$ if and only if $X' \overset{*}{\Longrightarrow} t$. The proof is similar to the one in the previous lemma and so is omitted.

Example 2. Consider the automaton M of Example III.1.1 and the grammar G of Example 1 in this section. M was obtained from G by the process used in Lemma 2. We could easily get another grammar G' to generate the family of sets recognized by M by applying the method of Lemma 3 to the deterministic automaton pM.

Theorem 4. Let $a \in \Omega_{\langle \lambda, i \rangle}$ and $V \subseteq (W_\Omega)_i$ be a recognizable set. Then V^{*a} is a recognizable set. If U is a recognizable family of sets, then so is $U \cdot_a V$. (We have again extended the $\cdot_a$ operation from sets to indexed families of sets.)

(ii) $P' = \{A \to \pi_{\langle w,i \rangle}(f)A_o \ldots A_{n-1} | f \in \Omega_{\langle w,i \rangle}$ and $A \to fA_o \ldots A_{n-1}$ is in $P\}$;

(iii) $Z' = Z$.

Then $\bar{\pi}(U) = L(G)$.

Let $G = \langle \Delta, \Sigma, P, Z \rangle$ be a grammar for V. Define $G' = \langle \Omega, \Sigma', P', Z' \rangle$ as follows:

(i) $\Sigma' = N \bigcup \Omega$

(ii) $P' = \{A \to fA_o \ldots A_{n-1} | f \in \Omega_{\langle w,i \rangle}$ and $A \to \pi_{\langle w,i \rangle}(f)A_o \ldots A_{n-1}$ is in $P\}$;

(iii) $Z' = Z$.

Then $L(G') = \bar{\pi}^{-1}(V)$. This works because each $f \in \Delta_{\langle w,i \rangle}$ has only a finite number of pre-images in $\Omega_{\langle w,i \rangle}$. Thus P' remains finite.

Remark. Consider the following class of semi-Thue grammars: Each $G = \langle \Omega, \Sigma, P, Z \rangle$ in the class is such that $N_{\langle w,i \rangle} = \phi$ for $w \neq \lambda$ and each $p \in P$ is of the form $t \to t'$ with $t, t' \in (W_\Sigma)_i$. The reader can easily satisfy himself that this class of grammars (restricted to the non-sorted case) is equivalent to the class of regular systems defined in Brainerd (1).

Let L(G) be the indexed family of sets generated by the regular system G. It can be shown that to each regular system G, we can effectively associate a regular grammar G' such that L(G) = L(G'). (The proof is a simple generalization of the method used by Brainerd.) A more important result is:

Theorem 6 (Büchi). Let U be recognizable. One can effectively find a regular system $G = \langle \Omega, \Omega, P, Z \rangle$ such that L(G) = U. That is, recognizable (indexed families of) sets can be generated by regular systems without any non-terminals.

The proof of the above is again a simple generalisation of results in Brainerd (1) and is motivated by the results of Büchi (3).

III.3. Regular Sets

Let Σ be a finite set. One of the most important notions in conventional theory is that of 'regular set of strings'. The class of regular sets of strings, over the alphabet Σ, is defined to be the least class of subsets of Σ^* which contains the finite subsets and is closed under the operations of union, complex product, nnd Kleene closure (see Kleene). A very important property of this class is that it is the same as the class of recognizable sets over Σ(and thus the same as the class of sets generated by regular grammars over Σ). We wish to generalise the concept 'regular' to indexed families of sets (over a many-sorted alphabet) and prove an analogous equivalence theorem.

Unfortunately, the concept does not generalise 'regularly'. For an indication of why this is so we refer the reader to Thatcher and Wright. Let Σ be a many-sorted alphabet. We proceed by defining the class of Σ-regular sets to be the least class of sets in $\hat{W}_\Sigma$ which contains the finite subsets of each sort and is closed under the operations of union (restricted to sets of the same sort), a-complex product and a-Kleene closure for each individual symbol a in Σ. This definition in fact restricts Σ-regular sets to be subsets of $(W_\Sigma)_i$ for some $i \in I$.

$U \sqsubseteq (W_\Sigma)_i$, some $i \in I$, is regular (over Σ) if there exists a finite many-sorted alphabet Σ' such that $\Sigma'_{\langle w,j \rangle} = \Sigma_{\langle w,j \rangle}$ for $w \neq \lambda$ and U is Σ'-regular. An indexed family of sets U is regular (over Σ) if each U_i is regular. We state without proof the following classical result (the proof is exactly analogous to that found in Thatcher and Wright or the improved proof in Arbib and Give'on):

Theorem 1: U is recognizable if and only if it is regular.

Corollary. Let Σ be a many-sorted alphabet. The following are equivalent classes of indexed families of sets:

(i) The class of indexed families of sets recognizable by Σ-automata;

(ii) The class of indexed families of sets generated by regular grammars over Σ;

(iii) The class of regular indexed families of sets over Σ.

III.4 Equational Sets

In this section, we present a generalisation of the work reported in Mezei and Wright. The material, except for a few refinements necessary for the many-sorted case, is basically the same as in the above and we try to indicate this with our notation. The reader who is not familiar with the subject matter is referred to the above for the details of proofs and constructions.

Let Σ be a many-sorted alphabet and A a Σ-automaton. We note the following property of each f_{pA}, $f \in \Sigma_{\langle w,i \rangle}$: Given arbitrary families of sets $S_{w_j} \subseteq (pA)_{w_j}$ for $0 \leq j \leq n-1$, then

$$f_{pA}(\bigcup_{S_{w_o}} R_o, \ldots, \bigcup_{S_{w_{n-1}}} R_{n-1}) = \bigcup_{S_{w_o}} \ldots \bigcup_{S_{w_{n-1}}} f_{pA}(R_o, \ldots, R_{n-1}) \text{ for}$$

$R_j \in S_{w_j}$, $0 \leq j \leq n-1$. We say that each f_{pA} has the property of being completely distributive.

Let $X = \{X_i\}_{i \in I}$ be an indexed family of sets called variables. Let I be ordered (by enumeration) and let each X_i be ordered (by enumeration). A term $t \in \hat{W}_\Sigma(X)$ is linear if each variable in t occurs no more than once. A term is linear distinguished if the leftmost occurrence of a variable of a given sort $i \in I$ is $x_o \in X_i$, the next occurrence of a variable of sort $i \in I$ is $x_1 \in X_i$, and so on. Terms in $\hat{W}_\Sigma$ (that is, terms without variables) are linearly distinguished. All terms t may be associated with a unique linear distinguished term $\tilde{t}$ from which t may again be obtained by a substitution of variables. Let $t \in \hat{W}_\Sigma(X)$. We let Σ_t and $\mathcal{U}_t$ denote the collection of operator symbols and variables, respectively, which occur in t.

Since $\mathcal{U}_t$ inherits an order from X, t determines in a natural way (using the Fundamental Theorem of Algebra) a <u>term function</u> $|At|$: $pA_{i_o} X \ldots X\, pA_{i_{n-1}} \to pA_i$ where $\mathcal{U}_t$ is the ordered set $\{y_o, \ldots, y_{n-1}\}$, y_j is of sort X_{i_j} for $0 \leq j \leq n-1$, and t is of sort i. Note that $|At|$ may be obtained as an appropriate restriction of $|A\tilde{t}|$. In general, term functions are not completely distributive. They, however, have the following property, called <u>distributivity over ω-chains</u> (where an <u>ω-chain</u> over a set S is a non-decreasing sequence of subsets of S indexed by the natural numbers): Given $t \in \hat{W}_\Sigma(X), \mathcal{U}_t = \{x_o, \ldots, x_{n-1}\}$, and an ω-chain $S_o^j \subseteq S_1^j \subseteq \ldots$ over A_{i_j} $(x_j \in X_{i_j})$ for each $0 \leq j \leq n-1$, then $|At|(\bigsqcup_{k\in\underline{N}} S_k^o, \ldots, \bigsqcup_{k\in\underline{N}} S_k^{n-1}) = \bigsqcup_{k\in\underline{N}} \{|At|(S_k^o, \ldots, S_k^{n-1})\}$. In particular, term functions are <u>order preserving</u>. That is, given $S_j, R_j \subseteq A_{i_j}$ for $0 \leq j \leq n-1$, if $S_j \subseteq R_j$, then $|At|(S_o, \ldots, S_{n-1}) \subseteq |At|(R_o, \ldots, R_{n-1})$.

Let $E \subseteq W_\Sigma(X)_i$. Define $\Sigma_E = \bigsqcup_{t\in E} \Sigma_t$ and $\mathcal{U}_E = \bigsqcup_{t\in E} \mathcal{U}_t$. Again $\mathcal{U}_E$ inherits an ordering from X. We associate with E a mapping $|AE|: pA_{i_o} X \ldots X\, pA_{i_{n-1}} \to pA_i$ where $|AE|(S_o, \ldots, S_{n-1}) = \bigsqcup_{t\in E}\{|At|(S_o, \ldots, S_{n-1})\}$ for $S_j \subseteq A_{i_j}$, $0 \leq j \leq n-1$. Of course, i_j is the sort of y_j in the ordered set $\mathcal{U}_E = \{y_o, \ldots, y_{n-1}\}$. Note that since, for any $t' \in E$, y_j may not occur in t', strictly speaking the expression $|At'|(S_o, \ldots, S_{n-1})$ is incorrect. We extend our notation to allow such expressions and interpret the meaning to be that $|At'|$ is empty in the j'th argument.

A <u>system of equations</u> $\mathcal{E}$ is a finite sequence of expressions of the form $x_j = E_j$ (called an equation) for $0 \leq j \leq m-1$ where each x_j is a distinct variable (called the <u>left hand side</u> of the j'th equation) and each E_j (called the <u>right hand side</u> of the j'th equation) is a finite set of terms of the same sort as x_j. Let $\mathcal{V}_{\mathcal{E}}$ be the ordered set of variables of the system $\mathcal{E}$. $\mathcal{V}_{\mathcal{E}}$ can be ordered and the equations are ordered accordingly. Let $m = \{m_i\}_{i\varepsilon I}$ where m_i is the number of equations in $\mathcal{E}$ with left hand sides of sort i. We re-index the right hand sides of $\mathcal{E}$ as follows: For each $i \varepsilon I$, the ordered set of sets of terms are enumerated by $E_{i,o}, \ldots, E_{i,m_i-1}$ where $(x_{j_o}, \ldots, x_{j_{m_i-1}})$ is the ordered set of variables of type i in $\mathcal{V}_{\mathcal{E}}$. We say that we have a system $\mathcal{E}$ of $m = \{m_i\}$ equations. Define $E_i = \bigsqcup_{k=0}^{m_i-1} \{E_{i,k}\}$ and $\Sigma_{\mathcal{E}} = \bigsqcup_{i\varepsilon I} \Sigma_{E_i}$.

Let $\mathcal{V}_{\mathcal{E}} = \{y_o, \ldots, y_{n-1}\}$ and let $\{0, 1, \ldots, k\}$ be an ordering of I.

We can associate with $\mathcal{E}$ the function $|A\mathcal{E}|:(pA_o)^{m_o} X \ldots X(pA_k)^{m_k} \to (pA_o)^{m_o} X \ldots X (pA_k)^{m_k}$ where for all $S_\ell^j, R_\ell^j \subseteq W_\Sigma(X)_j$ $(0 \leq \ell \leq m_j-1$ and $0 \leq j \leq k)$ $|A\mathcal{E}|(S_o^o, \ldots, S_{m_o-1}^o, \ldots, S_o^k, \ldots, S_{m_k-1}^k) =$
$(R_o^o, \ldots, R_{m_o-1}^o, \ldots, R_o^k, \ldots, R_{m_k-1}^k)$ if and only if $R_\ell^j = |AE_{\ell,j}|$
$(S_o^o, \ldots, S_{m_o-1}^o, \ldots, S_o^k, \ldots, S_{m_k-1}^k)$ for each $0 \leq \ell \leq m_j-1$ and $0 \leq j \leq k$.

Let L be the lattice $(pA_o)^{m_o} X \ldots X (pA_k)^{m_k}$ ordered by componentwise set inclusion and denote by 0 the $(m_o + \ldots + m_k)$-tuple of empty sets. Since term functions are order preserving, $|A\mathcal{E}|$ is a monotonic map of L into itself and the sequence $\{|A\mathcal{E}|^j(0)\}$ is an ω-chain, where $|A\mathcal{E}|^j(0)$ is defined by $|A\mathcal{E}|^o(0) = 0$ and $|A\mathcal{E}|^{n+1}(0) = |A\mathcal{E}|^n(|A\mathcal{E}|(0))$. $|A\mathcal{E}|$ inherits from term functions the property of distrinbutivity over ω-chains and so we have

$$|A\mathcal{E}|(\bigsqcup_{j\in\underline{N}}\{|A\mathcal{E}|^j(0)\}) = \bigsqcup_{j\in\underline{N}}\{|A\mathcal{E}|(|A\mathcal{E}|^j(0))\}$$

$$= \bigsqcup_{j\in\underline{N}}\{|A\mathcal{E}|^j(0)\}.$$

We denote $\bigsqcup_{j\in\underline{N}}\{|A\mathcal{E}|^j(0)\}$ by $[A\mathcal{E}^\omega]$. Thus $[A\mathcal{E}^\omega]$ is a fixed point of $|A\mathcal{E}|$ and in fact, since 0 is less than or equal to any fixed point of $|A\mathcal{E}|$ and $|A\mathcal{E}|$ is monotonic, $[A\mathcal{E}^\omega]$ is the least fixed point of $|A\mathcal{E}|$.

Given a system $\mathcal{E}$ of m equations, we define the following properties for it:

(i) $\mathcal{E}$ is of depth one if and only if its terms are of depth one;

(ii) $\mathcal{E}$ is deterministic if and only if, for each $i \in I$, $\{E_{i,j}\}$ is a partition of the set of all terms t of sort i such that $d(t) = 1$ and $\mathcal{V}_t \subseteq \mathcal{V}_\mathcal{E}$;

(iii) $\mathcal{E}$ is reduced if and only if for all Σ-algebras A, all the components of $[A\mathcal{E}^\omega]$ are non-empty.

Theorem 1. There is an effective procedure which, when applied to a system $\mathcal{E}$, yields a system $\mathcal{G}$ with the properties:

(i) $\mathcal{G}$ is deterministic

(ii) $\mathcal{G}$ is reduced, and

(iii) For any Σ-algebra A, any component of $[A\mathcal{E}^\omega]$ is a union over some components of $[A\mathcal{G}^\omega]$ (of the same sort).

We obtain the above system $\mathcal{G}$ by first reducing $\mathcal{E}$ to a system $\mathcal{F}$ of depth one (see Corollary to Theorem IV.13). With $\mathcal{F}$ we associate a deterministic system $\mathcal{F}'$ by a subset construction similar to the method used for obtaining a deterministic automaton from a non-deterministic one. We then associate a reduced system $\mathcal{G}$ with $\mathcal{F}'$ by, basically, eliminating the use of variables which are not 'used' in the function $|A\mathcal{F}'|$.

We are now ready to apply these results to the case when the Σ-algebra under consideration is W_Σ. A simple result that follows immediately from the above is:

Lemma 2. If $\mathcal{E}$ is an equational system that is both deterministic and reduced, then the family of collections of components of each sort of $[W_\Sigma \mathcal{E}^\omega]$ is a finite congruence on W_Σ.

The following is the main result of this section:

Theorem 3. Given an equational system $\mathcal{E}$, each indexed (by I) family of components of $[W_\Sigma \mathcal{E}^\omega]$ is recognizable. Conversely, given a family of sets U recognizable over Σ, there exists an equational system $\mathcal{G}$ such that U is an indexed family of components of $[W_\Sigma \mathcal{G}^\omega]$.

Proof. Given $\mathcal{E}$, we can obtain a reduced, deterministic system $\mathcal{G}$ such that a component of $[W_\Sigma \mathcal{E}^\omega]$ is a union of components of $[W_\Sigma \mathcal{G}^\omega]$. By the previous lemma, the family of collections of components of $[W_\Sigma \mathcal{G}^\omega]$ of the same sort is a finite congruence on W_Σ. Hence each component of $[W_\Sigma \mathcal{E}^\omega]$ is recognizable.

Conversely let U be recognizable. There is a regular grammar $G = \langle \Sigma, \Delta, P, Z \rangle$ such that $L(G) = U$. Form the equation system $\mathcal{G}$ as follows:

(i) Let $\mathcal{V}_{\mathcal{G}}$ be a set of variables in one-one correspondence to the disjoint union of N;

(ii) For each $A \in N_{\langle \lambda, i \rangle}$, let the right hand side of the equation with left hand side χ_A (corresponding to A, according to (9)) be $E_{i,A} = \{t' \mid A \to t$ is in P and t' is obtained from t by replacing non-terminals by the corresponding variables$\}$.

We claim that L(G) is the indexed family of components of $[W_\Sigma \mathcal{G}^\omega]$ corresponding to the variables associated with Z. The proof is omitted.

Example 1. Given the alphabet Σ of Example III.2.1, consider the following equational system $\mathcal{E}$ of m = {2, 6} equations

$$\chi_Z = \{\lambda, {}^*\chi_F\chi_E, {}^*\chi_G\chi_E\};$$

$$\chi_E = \{\lambda\};$$

$$\chi_A = \{a\};$$

$$\chi_B = \{b\};$$

$$\chi_C = \{+\chi_A\chi_D, +\chi_A\chi_B\};$$

$$\chi_D = \{+\chi_C\chi_B\};$$

$$\chi_F = \{+\chi_A\chi_B\};$$

$$\chi_G = \{+\chi_A\chi_D\}.$$

It should be clear that the component of $[W_\Sigma \mathcal{E}^\omega]$ corresponding to χ_Z is $\{\lambda, {}^*{+}ab\lambda, {}^*{+}a{+}{+}abb\lambda, \text{etc.}\} = L(G)$.

If we consider the second half of the example III.2.1 and modify the above system $\mathcal{E}$ so that m = {2, 7} and we insert the equation $\chi'_Z = \{+\chi_A\chi_B, +\chi_A\chi_D\}$ after the equation $\chi_E = \{\lambda\}$, then it is clear that the family of components of $[W_\Sigma \mathcal{E}^\omega]$ corresponding to $\{\chi_Z, \chi'_Z\}$ is L(G).

Let A be any Σ-algebra and $C \subseteq A$ an indexed family of sets. C is <u>equational</u> with respect to A if and only if there exists a system of equations $\mathcal{E}$ such that C is an indexed family of components of $[A\mathcal{E}^\omega]$. The following is the most important theorem of this chapter.

<u>Theorem 4 (Equivalence)</u>. Let Σ be a many-sorted alphabet. The following classes of indexed families of sets of terms are equivalent:

(i) The class of recognizable (families of) sets of terms over Σ;

(ii) The class of (families of) sets of terms generated by regular grammars over Σ;

(iia) The class of (families of) sets of terms generated by regular systems (with or without any non-terminals) over Σ;

(iii) The class of (families of) regular sets of terms over Σ;

(iv) The class of equational (families of) sets of terms over Σ.

Proof. A simple consequence of Theorems III.3.1 and III.4.3 and the Remark at the end of Section 2.

IV CONTEXT FREE SETS

IV.0 Introduction

In the previous chapter we generalised a 'natural' class of sets to the many-sorted case. These were the recognizable/regular/equational sets. We now proceed to generalise the next 'natural' class of sets to the many-sorted case and study the class of grammars which generate them. These are the context free sets and, as we have seen in Chapter II, the context free grammars form a proper sub-class of the class of semi-Thue grammars.

We proceed by obtaining a series of results leading to a Normal Form theorem for context free grammars exactly analogous to Chomsky normal form grammars for context free sets of strings (see Chomsky (1)). Along the way we also obtain a theorem on canonic systems of derivation. These are based on results obtained in Fischer (1) for macro-grammars. We then proceed to prove a number of classical results about context free grammars. Where appropriate (see the Preface), the reader is referred to proofs already published (for the generalised (non-sorted) case) in the literature.

IV.1. Context Free Grammars

We begin with a few examples of context free grammars.

Example 1. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $I = \{0, 1\}$;

(ii) $\Omega_{\langle \lambda,o \rangle} = \{\lambda\}$, $\Omega_{\langle \lambda,1 \rangle} = \{a, b, c\}$, $\Omega_{\langle 11,1 \rangle} = \{f\}$, $\Omega_{\langle 1o,o \rangle} = \{h\}$, $\Omega_{\langle 111,1 \rangle} = \{g\}$;

(iii) $N_{\langle \lambda,o \rangle} = \{Z\}$, $N_{\langle 111,1 \rangle} = \{F\}$;

(iv) $P = \{Z \to h(Fabc, \lambda), F(x, y, z) \to F(f(a, x), f(b, y), f(c, z)),$
$F(x, y, z) \to gxyz\}$ where $x, y, z \in X_1$.

The set generated by G is $L(G) = \{hgabc\lambda, hgfaafbbfcc\lambda, hgfafaafbfbbfcfcc\lambda,$ etc.$\}$.

If we change the above grammar by adding $Z \in N_{\langle \lambda,1 \rangle}$ to Σ and $Z \to Fabc$ to P then L(G) is the indexed family of sets $\{\{hgabc\lambda, hgfaafbbfcc\lambda,$ etc.$\}, \{gabc, gfaafbbfcc, gfafaafbfbbfcfcc,$ etc.$\}\}$.

Example 2. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $I = \{0, 1\}$;

(ii) $\Omega_{\langle \lambda,o \rangle} = \{\lambda\}$, $\Omega_{\langle \lambda,1 \rangle} = \{a\}$, $\Omega_{\langle 10,1 \rangle} = \{*\}$, $\Omega_{\langle 11,1 \rangle} = \{+\}$;

(iii) $N_{\langle \lambda, 0 \rangle} = \{Z\}$, $N_{\langle 1,1 \rangle} = \{B\}$;

(iv) $P = \{Z \to *a\lambda, Z \to *B(a)\lambda, B(x) \to B(+xx), B(x) \to +xx\}$
where $x \in X_1$.

The set generated by G is $L(G) = \{*a\lambda, *{+}aa\lambda, *{+}{+}aa{+}aa\lambda,$ etc.$\}$

If we change the above grammar by adding $Z \in N_{\langle \lambda,1 \rangle}$ to Σ and $Z \to B(a)$, $Z \to a$ to P then L(G) is the indexed family of sets $\{\{*a\lambda, *{+}aa\lambda, *{+}{+}aa{+}aa\lambda,$ etc.$\}, \{a, +aa, ++aa+aa,$ etc.$\}\}$. Note that the second component of L(G) is the set of balanced binary trees on the symbols + and a with the leaves labelled by a only and the internal nodes by + only.

Example 3. Here we give two examples over non-sorted alphabets closely related to the above examples (see Example I.4.3 for the explanation of the notation):

(a) Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $\Omega_o = \{a\}$, $\Omega_2 = \{+\}$;

(ii) $N_o = \{Z\}$, $N_1 = \{B\}$;

(iii) $P = \{Z \to a,\ Z \to B(a),\ B(x) \to B(+xx),\ B(x) \to +xx\}$.

Then G generates the set L(G) = {a, +aa, ++aa+aa, etc.}

(b) Let $G = \langle \Omega, \Sigma, P, Z \rangle$ where:

(i) $\Omega_o = \{a, b, c\}$, $\Omega_2 = \{f\}$, $\Omega_3 = \{g\}$;

(ii) $N_o = \{Z\}$, $N_3 = \{F\}$;

(iii) $P = \{Z \to Fabc,\ F(x, y, z) \to F(f(a, x), f(b, y), f(c, z)),$
$F(x, y, z) \to gxyz\}$.

Then the set generated by G is L(G) = {gabc, gfaafbbfcc, gfafaafbfbbfcfcc, etc.}.

We note that in the above grammars there was no restriction put on where a non-terminal may be in a term before we can apply a production. This corresponds to the notion of unrestricted derivation in the grammars for context free sets of strings. Now, in the conventional case, we have a notion of left-most derivation for context free grammars. Analogously, we define the concept of top-down or outside-in (OI) derivation for context free term grammars: An outside-in derivation is one in which productions are applied only to top level symbols in N, where a symbol $F \in N$ is said to be at top level in a term t if it does not occur in the argument list of any $H \in N$.

In the conventional context free case, to every unrestricted derivation there corresponds a left-most derivation (with possibly a greater number of steps) in the same grammar which generates the same string. We will prove an analogous result for context free term grammars. In fact, the conventional theorem becomes a special case of the more general result (see Remark at the end of Chapter II).

Remark. We note that another mode of derivation may be distinguished for term grammars. An inside-out (IO) or bottom-up derivation is one in which productions are applied to a non-terminal $F \in N$ if and only if no non-terminal $H \in N$ occurs in the argument list of F. In the string case, this would correspond to a right-most derivation. The expected result, in analogy with the conventional case, is that to any unrestricted derivation for a term t, there corresponds a bottom-up derivation (with possibly a greater number of steps) in the same grammar which generates the same string. That this is not so is a consequence of the fact that terms are not symmetric in the same sense that strings are. (A string 'read backwards' is still a string but a term 'read backwards' is not necessarily a term.) See Fischer (1), (2) for a similar distinction. We will not discuss IO derivations further here except to note that the class of sets of terms generated by context free grammars using only IO derivations is not the same as the class of sets of terms generated by context free grammars using only OI (or unrestricted) derivations.

Lemma 1 (Parallel Derivation). Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a context free term grammar and allow only OI derivations. Suppose $s, t \in \hat{W}_\Sigma$ such that $t \overset{*}{\Longrightarrow} s$ in p steps (which we write as $t \overset{p}{\Longrightarrow} s$). Let $r \in \hat{W}_\Sigma(X_w)$ and $(t_o, \ldots, t_{n-1}) \in (W_\Sigma)^w$ such that $t_o, \ldots, t_{n-1} \leq t$ and $\mathrm{Sub}_w(r; t_o, \ldots, t_{n-1}) = t$. Then there exists a term $t' \in \hat{W}_\Omega(X_w)$ and non-negative integers p_1 and p_2 such that $r \overset{p_1}{\Longrightarrow} t'$, $\mathrm{Sub}_w(t'; t_o, \ldots, t_{n-1}) \overset{p_2}{\Longrightarrow} s$, and $p_1 + p_2 = p$. (That is, we can first apply productions to r and eliminate all non-terminals from it to obtain t' and then proceed with applying productions to occurrences of $t_o, \ldots, t_{n-1}$ in $\mathrm{Sub}_w(t'; t_o, \ldots, t_{n-1})$ to get the results and not take any more steps in this derivation than in the original.)

Proof. Let $G, p, t, s, r, n, t_o, \ldots, t_{n-1}$ be given as in the statement of the theorem. The proof will be by induction on p.

$p = 0$: Obviously true.

$p > 0$: $t \Longrightarrow t'' \overset{p-1}{\Longrightarrow} s$. Suppose that $t \Longrightarrow t''$ via the production $F(x_o, \ldots, x_{k-1}) \rightarrow \nu$ in P for $F \in N_{\langle v,i \rangle}$, $\{x_o, \ldots, x_{k-1}\} = X_v$, and $\nu \in W_\Sigma(X_v)_i$. Then $t = Sub_i(\tilde{t}; F\xi_o \ldots \xi_{k-1})$ and $t'' = Sub_i(\tilde{t}; Sub_v(\nu; \xi_o, \ldots, \xi_{k-1}))$ for some $\tilde{t} \in \hat{W}_\Sigma(X_i)$ and $(\xi_o, \ldots, \xi_{k-1}) \in (W_\Sigma)^v$.

Case 1: The transformed occurrence of $F\xi_o \ldots \xi_{k-1}$ in t is not contained in any occurrence of t_j (which is replaced by x_{j,w_j} in r) for $0 \leq j \leq n-1$. Then there exists $\tilde{r} \in \hat{W}_\Sigma(X_w, X_i)$ and $(\overline{\xi}_o, \ldots, \overline{\xi}_{k-1}) \in W_\Sigma(X_w)^v$ such that $r = Sub_i(\tilde{r}; F\overline{\xi}_o \ldots \overline{\xi}_{k-1})$, $\tilde{t} = Sub_v(\tilde{r}; t_o, \ldots, t_{k-1})$, and $\xi_j = Sub_v(\overline{\xi}_j; t_o, \ldots, t_{k-1})$ for $0 \leq j \leq k-1$. Let $r' = Sub_i(\tilde{r}; Sub_v(\nu; \overline{\xi}_o, \ldots, \overline{\xi}_{k-1}))$. We have:

(i) $t = Sub_i(\tilde{t}; F\xi_o \ldots \xi_{k-1})$ implies $t'' = Sub_i(\tilde{t}; Sub_v(\nu; \xi_o,\ldots,\xi_{k-1}))$

(ii) $r = Sub_i(\tilde{r}; F\overline{\xi}_o \ldots \overline{\xi}_{k-1})$ implies $r' = Sub_i(\tilde{r}; Sub_v(\nu; \overline{\xi}_o,\ldots,\overline{\xi}_{k-1})$;

and

(iii) $Sub_w(r'; t_o, \ldots, t_{n-1}) = t''$.

By the induction hypothesis, there exists $t' \in \hat{W}_\Omega(X_w)$ such that $r' \overset{q_1}{\Longrightarrow} t'$, $Sub_w(t'; t_o, \ldots, t_{n-1}) \overset{q_2}{\Longrightarrow} s$ and $q_1+q_2 = p-1$. But then $r \Longrightarrow r' \overset{q_1}{\Longrightarrow} t'$, so we are done if we take $p_1 = q_1+1$, $p_2 = q_2$.

Case 2: The transformed occurrence of $F\xi_o \ldots \xi_{k-1}$ in t is contained in some occurrence of t_j (which is replaced by x_{j,w_j} in r) for some $0 \leq j \leq n-1$. Since $F\xi_o \ldots \xi_{k-1}$ is the term that is transformed in the first step of

$t \overset{p}{\Longrightarrow} s$, it must occur at the top level of t and so at the top level of t_j. Then there exists $\tilde{t}_j \in W_\Sigma(X_i)_{w_j}$ such that $t_j = \mathrm{Sub}_i(\tilde{t}_j; F\xi_o \ldots \xi_{k-1})$. Thus $t = \mathrm{Sub}_{w_j}(\tilde{t}; t_j) = \mathrm{Sub}_{w_j}(\tilde{t}; \mathrm{Sub}_i(\tilde{t}_j; F\xi_o \ldots \xi_{k-1}))$ for $\tilde{t} \in \hat{W}_\Sigma(X_{w_j})$ and $r = \mathrm{Sub}_i(\tilde{r}; x_{i,o})$ where $\mathrm{Sub}_w(\tilde{r}; t_o, \ldots, t_{n-1}) = \tilde{t}$. Let $t_j' = \mathrm{Sub}_i(\tilde{t}_j; \mathrm{Sub}_v(\nu; \xi_o, \ldots, \xi_{k-1}))$. Then $t_j \Longrightarrow t_j'$ via the rule $F(x_o, \ldots, x_{k-1}) \to \nu$ in P. Also, $t'' = \mathrm{Sub}_{w_j}(\tilde{t}; \mathrm{Sub}_i(\tilde{t}_j; \mathrm{Sub}_v(\nu; \xi_o, \ldots, \xi_{k-1}))) = \mathrm{Sub}_{w_j}(\tilde{t}; t_j')$. Let $r'' = \mathrm{Sub}_{w_j}(\tilde{r}; y)$, $\{y\} = Y_{w_j}$. It is clear that $t'' = \mathrm{Sub}_{w_j}(\mathrm{Sub}_w(r''; t_o, \ldots, t_{n-1}); t_j')$.

By the induction hypothesis, there exists $\tilde{t}' \in \hat{W}_\Omega(X_w, Y_{w_j})$ such that $r'' \overset{q_1}{\Longrightarrow} \tilde{t}'$, $\mathrm{Sub}_{w_j}(\mathrm{Sub}_w(\tilde{t}'; t_o, \ldots, t_{n-1}); t_j') \overset{q_2}{\Longrightarrow} t'$ and $q_1+q_2 = p-1$. Since r'' contains exactly one occurrence of y, so does $\tilde{t}'$. Thus, since $t_j \Longrightarrow t_j'$, then $\mathrm{Sub}_{w_j}(\mathrm{Sub}_w(\tilde{t}'; t_o, \ldots, t_{n-1}); t_j) \Longrightarrow \mathrm{Sub}_{w_j}(\mathrm{Sub}_w(\tilde{t}'; t_o, \ldots, t_{n-1}); t_j')$. Let $t' = \mathrm{Sub}_{w_j}(\tilde{t}'; x_{j,w_j})$. Since $r = \mathrm{Sub}_{w_j}(r''; x_{j,w_j})$, $r \overset{q_1}{\Longrightarrow} t'$ by simply substituting x_{j,w_j} for y in each step of the derivation $r'' \overset{q_1}{\Longrightarrow} \tilde{t}'$. t' satisfies the theorem since:

$$\mathrm{Sub}_w(t'; t_o, \ldots, t_{n-1}) = \mathrm{Sub}_{w_j}(\mathrm{Sub}_w(\tilde{t}'; t_o, \ldots, t_{n-1}); t_j)$$
$$\Longrightarrow \mathrm{Sub}_{w_j}(\mathrm{Sub}_w(\tilde{t}'; t_o, \ldots, t_{n-1}); t_j') \overset{q_2}{\Longrightarrow} s \quad .$$

Let $p_1 = q_1$ and $p_2 = q_2 + 1$.

Suppose G is a context free grammar. We will denote the (indexed family of) set(s) generated by using only outside-in derivations in G by $L_{OI}(G)$. Similarly, $L_U(G)$ will be used to denote the (indexed family of) set(s) generated by using unrestricted derivations in G. Also, $\underset{U}{\Longrightarrow}$ and

$\Longrightarrow$ and $\underset{OI}{\Longrightarrow}$ will be used to denote unrestricted and outside-in derivations, respectively, in G.

<u>Theorem 2 (Canonic Derivations).</u> $L_{OI}(G) = L_U(G)$ for any context free term grammar G.

<u>Proof</u>. Let $G = \langle \Omega, \Sigma, P, Z \rangle$. It is clear that $L_{OI}(G) \subseteq L_U(G)$. To prove the opposite relation, suppose $t_o \overset{p}{\underset{U}{\Longrightarrow}} s$ ($t_o \in \hat{W}_\Sigma$, $s \in \hat{W}_\Omega$). We will show by induction on p that $t_o \overset{*}{\underset{OI}{\Longrightarrow}} s$; that is, we can use a canonical derivation at the expense of perhaps using more derivation steps than in the non-canonical derivation.

p = 0: Obvious.

p > 0: $t_o \underset{U}{\Longrightarrow} t_1 \overset{p-1}{\underset{U}{\Longrightarrow}} s$, where $t_o \underset{U}{\Longrightarrow} t_1$ by means of the rule $F(x_o, \ldots, x_{n-1}) \to \nu$ in P for $\{x_o, \ldots, x_{n-1}\} = X_w$, $\nu \in W_\Sigma(X_w)_i$, and $F \in N_{\langle w,i \rangle}$. Let $F\xi_o \ldots \xi_{n-1}$ be the subterm of t_o that is rewritten and let $t'_1 \in \hat{W}_\Sigma(X_i)$ such that $Sub_i(t'_1; F\xi_o \ldots \xi_{n-1}) = t_o$ and $Sub_i(t_1'; Sub_w(\nu; \xi_o \ldots \xi_{n-1})) = t_1$. By the induction hypothesis, $t_1 \overset{*}{\underset{OI}{\Longrightarrow}} s$. By the previous lemma, there exists $t' \in \hat{W}_\Omega(X_i)$ such that $t_1' \overset{*}{\underset{OI}{\Longrightarrow}} t'$ and $Sub_i(t'; Sub_w(\nu; \xi_o, \ldots, \xi_{n-1})) \overset{*}{\underset{OI}{\Longrightarrow}} s$. Now $Sub_i(t'; F\xi_o \ldots \xi_{n-1}) \overset{m}{\underset{OI}{\Longrightarrow}} Sub_i(t'; Sub_w(\nu; \xi_o, \ldots, \xi_{n-1})) \overset{*}{\underset{OI}{\Longrightarrow}} s$ where m is the number of occurrences of $x_{i,o}$ in t'. Substituting $F\xi_o \ldots \xi_{n-1}$ for $x_{i,o}$ in each step of the derivation $t_1' \overset{*}{\underset{OI}{\Longrightarrow}} t'$ gives $Sub_i(t_1', F\xi_o \ldots \xi_{n-1}) \overset{*}{\underset{OI}{\Longrightarrow}} Sub_i(t'; F\xi_o \ldots \xi_{n-1})$.

Putting the above together we get

$$t_o = \mathrm{Sub}_i(t_1'; F\xi_o \ldots \xi_{n-1})$$

$$\overset{*}{\underset{OI}{\Longrightarrow}} \mathrm{Sub}_i(t'; F\xi_o \ldots \xi_{n-1})$$

$$\overset{m}{\underset{OI}{\Longrightarrow}} \mathrm{Sub}_i(t'; \mathrm{Sub}_w(\nu; \ \xi_o, \ldots, \xi_{n-1}))$$

$$\overset{*}{\underset{OI}{\Longrightarrow}} s \quad .$$

Thus $L_U(G) \sqsubseteq L_{OI}(G)$ and we are done.

Given a context free term grammar $G = \langle \Omega, \Sigma, P, Z \rangle$, G is said to be in <u>standard form</u> if all derivations are OI and every production pin P is in one of the following forms:

(i) $F(x_o, \ldots, x_{n-1}) \to G(H_o(x_o,\ldots,x_{n-1}),\ldots,H_{m-1}(x_o,\ldots,x_{n-1}))$

where $F \in N_{\langle w,i \rangle}$, $G \in N_{\langle v,i \rangle}$, and, for each $0 \leq j \leq m-1$, $H_j \in N_{\langle w,v_j \rangle}$ for $v \neq \lambda$;

(ii) $F(x_o, \ldots, x_{n-1}) \to t$ where $F \in N_{\langle w,i \rangle}$ and $t \in W_\Omega(X_w)_i$.

<u>Theorem 3</u>. For every OI grammar G (that is, a context free grammar G using only OI derivations), we can effectively find another grammar G' in standard form so that $L_{OI}(G) = L_{OI}(G')$.

<u>Proof</u>. From $G = \langle \Omega, \Sigma, P, Z \rangle$ we will form a sequence of OI grammars $G = G^o, G^1, \ldots, G^q = G'$, $G^j = \langle \Omega, \Sigma^j, P^j, Z \rangle$ for $0 \leq j \leq q$, so that $L(G^j) = L(G)$, each G^j is a step on the way to putting G in standard form, and $G^q = G'$ is in standard form. (We drop the OI from $L_{OI}(G^j)$ when it is clear from the context that this is what we mean.)

Let $G^1 = \langle \Omega, \Sigma^1, P^1, Z \rangle$ where:

(i) $N^1_{\langle w,i \rangle} = N_{\langle w,i \rangle} \bigcup \{C_f | f \in \Omega_{\langle w,i \rangle}\}$ for all $w \neq \lambda$

and $N^1_{\langle \lambda,i \rangle} = N_{\langle \lambda,i \rangle}$ for all $i \in I$;

(ii) $P^1 = P \bigcup \{C_f(x_o, \ldots, x_{n-1}) \rightarrow f(x_o, \ldots, x_{n-1}) | f \in \Omega_{\langle w,i \rangle}$,

$w \neq \lambda$ and $\{x_o, \ldots, x_{n-1}\} = X_w\}$.

We apply one of the following two transformations to obtain G^{k+1} from G^k, if possible. If neither is applicable, G^k is in standard form.

(T_1) Transformation 1: If $F(x_o, \ldots, x_{n-1}) \rightarrow ft_o \ldots t_{m-1}$ is in P^k and $ft_o \ldots t_{m-1} \notin \hat{W}_\Omega(X_w)$ ($\{x_o, \ldots, x_{n-1}\} = X_w$), then replace that production by the production $F(x_o, \ldots, x_{n-1}) \rightarrow C_f(t_o, \ldots, t_{m-1})$. Then $N^{k+1} = N^k$ and $P^{k+1} = (P^k - \{F(x_o, \ldots, x_{n-1}) \rightarrow ft_o \ldots t_{m-1}\}) \bigcup \{F(x_o, \ldots, x_{n-1}) \rightarrow C_f(t_o, \ldots, t_{m-1})\}$.

(T_2) Transformation 2: If $F(x_o, \ldots, x_{n-1}) \rightarrow G(t_o, \ldots, t_{m-1})$ is in P^k and, for some $0 \leq j \leq m-1$, $t_j \neq H(x_o, \ldots, x_{n-1})$ for any $H \in N_{\langle w,i \rangle}$ ($\{x_o, \ldots, x_{n-1}\} = X_w$), then replace that production by the pair of productions $Q = \{F(x_o, \ldots, x_{n-1}) \rightarrow G(t_o, \ldots, t_{j-1}, W(x_o, \ldots, x_{n-1}), t_{j+1}, \ldots, t_{m-1}), W(x_o, \ldots, x_{n-1}) \rightarrow t_j\}$ where $W \notin N^k$ and $W \in N^{k+1}_{\langle w,v_j \rangle}$ (for $F \in N^k_{\langle w,i \rangle}$, $G \in N^k_{\langle v,i \rangle}$). Note that $t_j \in W_\Sigma(X_w)_{v_j}$.
Then $N^{k+1}_{\langle u,\ell \rangle} = N^k_{\langle u,\ell \rangle}$ for $\langle u,\ell \rangle \neq \langle w,v_j \rangle$ and $N^{k+1}_{\langle w,v_j \rangle} = N^k_{\langle w,v_j \rangle} \bigcup \{W\}$ while $P^{k+1} = (P^k - \{F(x_o, \ldots, x_{n-1}) \rightarrow G(t_o, \ldots, t_{m-1})\}) \bigcup Q$.

One of the above transformations applies to a grammar if and only if the grammar is not in standard form. To show that the sequence of grammars terminates (after a finite number of steps), we assign a non-negative integer to each grammar in the sequence and show that each transformation (strictly) reduces this number. Thus the number assigned to G^o serves as a bound on the length of the sequence.

Let M be the maximum rank of any $F \varepsilon N$. Define $\Phi:\underline{N} \times W_\Sigma(X) \to \underline{N}$ by:

(i) $\Phi_i(n, a) = 0$ if $a \varepsilon \Omega_{\langle\lambda,i\rangle} \bigcup X_i$;

(ii) $\Phi_i(n, ft_o \ldots t_{m-1}) = 1 + M + \sum_{j=0}^{m-1} \Phi_{w_j}(n, t_j)$ where $f \varepsilon \Omega_{\langle w,i\rangle}$ and $(t_o, \ldots, t_{m-1}) \varepsilon W_\Sigma(X)^w$;

(iii) $\Phi_i(n, G(t_o, \ldots, t_{m-1})) = |T| + \sum_{j\varepsilon T} \Phi_{w_j}(n, t_j)$ where $G \varepsilon N_{\langle w,i\rangle}$, $(t_o, \ldots, t_{m-1}) \varepsilon W_\Sigma(X)^w$, and $T = \{j | 0 \leq j \leq m-1$ and $t_j \neq H(x_o, \ldots, x_{n-1})$ for any $H \varepsilon N\}$.

Define $\Gamma:P \to \underline{N}$ by $\Gamma(F(x_o, \ldots, x_{n-1}) \to t) = \Phi_i(n, t)$ for $F \varepsilon N_{\langle w,i\rangle}$.
Define $\Delta:\{\text{grammars}\} \to \underline{N}$ by $\Delta(G) = \sum_{p\varepsilon P} \Gamma(p)$.

We now show that $\Delta(G^{k+1}) < \Delta(G^k)$ for each $1 \leq k \leq q - 1$.

<u>Case 1</u>. G^{k+1} is obtained from G^k by an application of T_1. Then we have:

(i) $\Gamma(F(x_o, \ldots, x_{n-1}) \to ft_o \ldots t_{m-1}) = 1 + M + \sum_{j=0}^{m-1} \Phi_{v_j}(n, t_j)$;

(ii) $\Gamma(F(x_o, \ldots, x_{n-1}) \to C_f(t_o, \ldots, t_{m-1})) \leq m + \sum_{j=1}^{m} \Phi_{v_j}(n, t_j)$;

(iii) $\Delta(G^k) - \Delta(G^{k+1}) = \Gamma(F(x_o, \ldots, x_{n-1}) \to ft_o \ldots t_{m-1}) - \Gamma(F(x_o, \ldots, x_{n-1}) \to C_f(t_o, \ldots, t_{m-1})) \geq M + 1 - m \geq 1$ (since $M \geq m$).

<u>Case 2</u>: G^{k+1} is obtained from G^k by an application of T_2. Then we have:

(i) $\Gamma(F(x_o, \ldots, x_{n-1}) \to G(t_o, \ldots, t_{m-1})) = |T| + \sum_{\ell \varepsilon T} \Phi_{v_\ell}(n, t_\ell)$;

(ii) $\Gamma(F(x_o, \ldots, x_{n-1}) \to G(t_o, \ldots, W(x_o, \ldots, x_{n-1}), \ldots, t_{m-1}))$

$= |T'| + \sum_{\ell \varepsilon T'} \Phi_{v_\ell}(n, t_\ell)$ where $T' = T - \{j\}$;

(iii) $\Delta(G^k) - \Delta(G^{k+1}) = |T| + \sum_{\ell \varepsilon T} \Phi_{v_\ell}(n, t_\ell) - (\Phi_{v_j}(n, t_j) + |T'| +$

$\sum_{\ell \varepsilon T'} \Phi_{v_\ell}(n, t_\ell)) = |T| - |T'| = 1.$

Thus we obtain a standard form grammar for G in at most $\Delta(G)$ steps. Obviously $L(G) = L(G^o) = L(G^1)$. We proceed to show that $L(G^{k+1}) \sqsubseteq L(G^k)$ for $1 \leq k \leq q - 1$:

Suppose t is a term of G^k and $t \overset{p}{\underset{G^{k+1}}{\Longrightarrow}} s$, $s \; \varepsilon \; \hat{W}_\Omega$. We show by induction on p that $t \overset{*}{\underset{G^k}{\Longrightarrow}} s$.

$p = 0$: Obvious.

$p > 0$: $t \underset{G^{k+1}}{\Longrightarrow} t' \overset{p-1}{\underset{G^{k+1}}{\Longrightarrow}} s$.

Case 1: $t \underset{G^{k+1}}{\Longrightarrow} t'$ by a rule which is also in P^k. Then t' is a term of G^k, so $t' \overset{*}{\underset{G^k}{\Longrightarrow}} s$ by the induction hypothesis. Hence $t \overset{*}{\underset{G^k}{\Longrightarrow}} s$.

Case 2: G^{k+1} is obtained from G^k by T_1 and $t \underset{G^{k+1}}{\Longrightarrow} t'$ by the rule $F(x_o, \ldots, x_{n-1}) \to C_f(t_o, \ldots, t_{m-1})$ which is in P^{k+1} but not in P^k. Then P^k has the rule $F(x_o, \ldots, x_{n-1}) \to ft_o \ldots t_{m-1}$. Thus

$t = Sub_i(\tilde{t};\ F\xi_o \ldots \xi_{n-1})$ and $t' = Sub_i(\tilde{t};\ Sub_w(C_f t_o \ldots t_{m-1};$ $\xi_o, \ldots, \xi_{n-1}))$ for some $\tilde{t}, \xi_o, \ldots, \xi_{n-1}$, and $F \in N^{k+1}_{\langle w,i \rangle}$. The only rule of P^{k+1} with C_f as the left part is $C_f(x_o, \ldots, x_{n-1}) \rightarrow fx_o \ldots x_{n-1}$, so $Sub_i(\tilde{t}\ ;\ Sub_w(ft_o \ldots t_{m-1};\ \xi_o, \ldots, \xi_{n-1})) \xrightarrow[G^{k+1}]{p-2} s$. By the induction hypothesis $Sub_i(\tilde{t};\ Sub_w(ft_o \ldots t_{m-1};\ \xi_o, \ldots, \xi_{n-1})) \xrightarrow[G^k]{*} s$.

Thus $t = Sub_i(\tilde{t};\ F\xi_o \ldots \xi_{n-1}) \xrightarrow[G^k]{} Sub_i(\tilde{t};\ Sub_w(ft_o \ldots t_{m-1};\ \xi_o,\ldots,\xi_{n-1}))$ $\xrightarrow[G^k]{*} s$.

Case 3: G^{k+1} is obtained from G^k by T_2. $t \xrightarrow[G^{k+1}]{} t'$ by the rule $F(x_o, \ldots, x_{n-1}) \rightarrow G(t_o, \ldots, W(x_o, \ldots, x_{n-1}), \ldots, t_{m-1})$ which is in P^{k+1} but not in P^k. By the construction, there is exactly one rule $W(x_o, \ldots, x_{n-1}) \rightarrow t_j$ in P^{k+1} for some term t_j and P^k has the rule $F(x_o, \ldots, x_{n-1}) \rightarrow G(t_o, \ldots, t_{m-1})$. Then $t = Sub_i(\tilde{t};\ F\xi_o \ldots \xi_{n-1})$ and $t' = Sub_i(\tilde{t};\ Sub_w(G(t_o, \ldots, W(x_o, \ldots, x_{n-1}), \ldots, t_{m-1});\ \xi_o, \ldots, \xi_{n-1}))$ for some $\tilde{t}, \xi_o, \ldots, \xi_{n-1}$ and $F \in N^{k+1}_{\langle w,i \rangle}$. Let $\bar{t}'$ be the result of replacing $W\xi_o \ldots \xi_{n-1}$ in t' by $y \in X_j$ ($W \in N^{k+1}_{\langle w,v_j \rangle}$) so $\bar{t}' =$ $Sub_i(\tilde{t};\ Sub_w(Gt_o \ldots, y, \ldots, t_{m-1};\ \xi_o, \ldots, \xi_{n-1}))$ and $t' =$ $Sub_{w_j}(\bar{t}', W\xi_o \ldots \xi_{n-1})$. By the parallel derivation lemma, there exists $r \in \hat{W}_\Omega(Y_{v_j})$ such that $\bar{t}' \xrightarrow[G^{k+1}]{p_1} r$, $Sub_{v_j}(r;\ W\xi_o \ldots \xi_{n-1}) \xrightarrow[G^{k+1}]{p_2} s$ and $p_1 + p_2 = p-1$. Since the only rule in P^{k+1} with W as the left part is $W(x_o, \ldots, x_{n-1}) \rightarrow t_j$ and since each occurrence of $W\xi_o \ldots \xi_{n-1}$ in $Sub_{v_j}(r;\ W\xi_o \ldots \xi_{n-1})$ is at top level, we can find a derivation

$$Sub_{v_j}(r;\ W\xi_o \ldots \xi_{n-1}) \xrightarrow[G^{k+1}]{q} Sub_{v_j}(r;\ Sub_w(t_j;\ \xi_o, \ldots, \xi_{n-1})) \xrightarrow[G^{k+1}]{p_2-q} s$$

and $p_1 + p_2 - q \leq p-1$. By the induction hypothesis $Sub_{v_j}(\bar{t}'; Sub_w(t_j; \xi_o, \ldots, \xi_{n-1})) \underset{G^k}{\overset{*}{\Longrightarrow}} s$. Also $F(x_o, \ldots, x_{n-1}) \to G(t_o, \ldots, t_{m-1})$ is in P^k. So $t = Sub_i(\tilde{t}; F\xi_o \ldots \xi_{n-1}) \underset{G^k}{\Longrightarrow} Sub_i(\tilde{t}; Sub_w(Gt_o \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1})) = Sub_{v_j}(\bar{t}'; Sub_w(t_j; \xi_o, \ldots, \xi_{n-1}) \underset{G^k}{\overset{*}{\Longrightarrow}} s$.

We now show the opposite relation $L(G^k) \sqsubseteq L(G^{k+1})$: Suppose $t \underset{G^k}{\overset{p}{\Longrightarrow}} s$ for some term t of G^k, $s \in \hat{W}_\Omega$. We show by induction on p that $t \underset{G^{k+1}}{\overset{*}{\Longrightarrow}} s$.

$p = 0$: obvious.

$p > 0$: $t \underset{G^k}{\Longrightarrow} t' \underset{G^k}{\overset{p-1}{\Longrightarrow}} s$.

Case 1: $t \underset{G^k}{\Longrightarrow} t'$ by a rule also in G^{k+1}. By the induction hypothesis

$t' \underset{G^{k+1}}{\overset{*}{\Longrightarrow}} s$. So $t \underset{G^{k+1}}{\overset{*}{\Longrightarrow}} s$.

Case 2: G^{k+1} is obtained from G^k by T_1 and $t \underset{G^k}{\Longrightarrow} t'$ by the rule $F(x_o, \ldots, x_{n-1}) \to ft_o \ldots t_{m-1}$ which is in P^k but not in P^{k+1}. Then P^{k+1} has the rule $F(x_o, \ldots, x_{n-1}) \to C_f(t_o, \ldots, t_{m-1})$. Thus $t = Sub_i(\tilde{t}; F\xi_o \ldots \xi_{n-1})$, $t_1 = Sub_i(\tilde{t}; Sub_w(ft_o \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1}))$ for some $\tilde{t}, \xi_o, \ldots, \xi_{n-1}$ and $F \in N^k_{\langle w,i \rangle}$. Then

$t \underset{G^{k+1}}{\Longrightarrow} Sub_i(\tilde{t}; Sub_w(C_f t_o \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1})) \underset{G^{k+1}}{\Longrightarrow} Sub_i(\tilde{t}; Sub_w(ft_o \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1}))$. By the induction hypothesis,

$t' \underset{G^{k+1}}{\overset{*}{\Longrightarrow}} s$ and so $t \underset{G^{k+1}}{\overset{*}{\Longrightarrow}} s$.

Case 3: G^{k+1} is obtained from G^k by T_2 and $t \underset{G^k}{\Longrightarrow} t'$ by the rule $F(x_o, \ldots, x_{n-1}) \to G(t_o, \ldots, t_{m-1})$ which is in P^k but not in P^{k+1}. Then P^{k+1} has the rules $F(x_o, \ldots, x_{n-1}) \to G(t_o, \ldots, t_{j-1}, W(x_o, \ldots, x_{n-1}), t_{j+1}, \ldots, t_{m-1})$ and $W(x_o, \ldots, x_{n-1}) \to t_j$ for some $0 \leq j \leq m-1$. We thus have $t = Sub_i(\tilde{t}; F\xi_o \ldots \xi_{n-1})$ and $t' = Sub_i(\tilde{t}; Sub_w(Gt_o \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1}))$ for some $\tilde{t}, \xi_o, \ldots, \xi_{n-1}$ and $F \in N^k_{\langle w,i \rangle}$. Then, $t \underset{U,G^{k+1}}{\Longrightarrow} Sub_i(\tilde{t}; Sub_w(G(t_o, \ldots, W(x_o, \ldots, x_{n-1}), \ldots, t_{m-1}); \xi_o, \ldots, \xi_{n-1})) \underset{U,G^{k+1}}{\Longrightarrow} Sub_i(\tilde{t}; Sub_w(Gt_o \ldots t_j \ldots t_{m-1}; \xi_o, \ldots, \xi_{n-1})) = t'$. By the induction hypothesis $t' \overset{*}{\underset{G^{k+1}}{\Longrightarrow}} s$ by an OI derivation and hence an unrestricted derivation.

Thus $t \overset{*}{\underset{UG^{k+1}}{\Longrightarrow}} s$. By the previous theorem, we can find an OI derivation of the same string. Thus $t \overset{*}{\underset{G^{k+1}}{\Longrightarrow}} s$.

Corollary: Suppose $\mathcal{E}$ is an equational system (as described in Section III.4) over the many-sorted alphabet Ω. If we consider an equation $x = E$ as representing a set of productions $\{x \to t | t \in E\}$ (with the variables considered as non-terminals), then we can reduce $\mathcal{E}$ to a system $\mathcal{F}$ of depth one by using only Transformation 2': If $A \to ft_o \ldots t_{m-1}$ is in P^k and, for some $0 \leq j \leq m-1$, $t_j \neq B \in N_{\langle \lambda, w_j \rangle}$ for $f \in \Omega_{\langle w,i \rangle}$, then replace that production by the pair of productions $Q = \{A \to ft_o \ldots t_{j-1}Wt_{j+1} \ldots t_{m-1}, W \to t_j\}$ where $W \notin N^k$. Then $N^{k+1}_{\langle \lambda, \ell \rangle} = N^k_{\langle \lambda, \ell \rangle}$ for $\ell \neq w_j$ and $N^{k+1}_{\langle \lambda, w_j \rangle} = N^k_{\langle \lambda, w_j \rangle} \bigcup \{W\}$ while $P^{k+1} = (P^k - \{A \to ft_o \ldots t_{m-1}\}) \bigcup Q$.

Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a context free term grammar. Then G is said to be in reduced form if all derivations are OI and every production p in P is in one of the following forms:

(i) $F(x_o, \ldots, x_{n-1}) \to t$ where $F \in N_{\langle w,i \rangle}$ and $t \in W_N(X_w)_i$;

(ii) $F(x_o, \ldots, x_{n-1}) \to fx_o \ldots x_{n-1}$ where $F \in N_{\langle w,i \rangle}$ and $f \in \Omega_{\langle w,i \rangle}$.

Theorem 4. For every OI grammar G, we can effectively find another OI grammar G' in reduced form so that L(G) = L(G').

Proof. Our reduced form is the generalization from non-sorted alphabets to many-sorted alphabets of Rounds' 'normal form'. The reader is referred to Rounds (1), (2), (3) for the proof and further discussion.

Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a context free term grammar. Then G is said to be in (Chomsky) Normal Form if all derivations are OI and every production $p \in P$ is in one of the following forms:

(i) $F(x_o, \ldots, x_{n-1}) \to G(H_o(x_o, \ldots, x_{n-1}), \ldots, H_{m-1}(x_o, \ldots, x_{n-1}))$

where $F \in N_{\langle w,i \rangle}$, $G \in N_{\langle v,i \rangle}$, each $H_j \in N_{\langle w,v_j \rangle}$ for $0 \leq j \leq m-1$ and $v \neq \lambda$;

(ii) $F(x_o, \ldots, x_{n-1}) \to fx_{j_o} \ldots x_{j_{k-1}}$ where $F \in N_{\langle w,i \rangle}$, $f \in \Omega_{\langle v,i \rangle}$ and $fx_{j_o} \ldots x_{j_{k-1}} \in W_\Omega(X_w)_i$;

(iii) $F(x_o, \ldots, x_{n-1}) \to x_j$ where $F \in N_{\langle w,i \rangle}$, with $w \neq \lambda$, and $0 \leq j \leq n-1$.

Note that these three types of productions are analogous to the three types of productions in a Chomsky normal form string grammar where the productions are of the form $A \to BC$, $A \to a$, or $A \to \lambda$ (A, B, C non-terminals, a terminal, λ the empty word).

Theorem 5 (Chomsky). For every OI grammar G, we can effectively find another OI grammar G' in normal form so that L(G) = L(G').

Proof. Apply Theorem 4 to G and Theorem 3 to the result.

Example 4. Let G be the context free grammar of Example 2. Then $G' = \langle \Omega, \Sigma', P', Z' \rangle$ is a normal form grammar such that $L(G) = L(G')$ where:

(i) $N'_{\langle \lambda,0 \rangle} = \{Z, L\}$, $N'_{\langle \lambda,1 \rangle} = \{A, C\}$, $N'_{\langle 1,1 \rangle} = \{B, D\}$, $N'_{\langle 10,0 \rangle} = \{S\}$;

(ii) $P' = \{Z \to S(A, L); Z \to S(C, L), C \to B(A), S(x, y) \to *xy, B(x) \to B(D(x)), B(x) \to +xx, D(x) \to +xx, A \to a, L \to \lambda\}$;

(iii) $Z' = Z$.

To obtain the normal form for the grammar of the second part of Example 2, we must add Z to $N'_{\langle \lambda,1 \rangle}$ above and $Z \to A$, $Z \to B(A)$ to P' above.

Example 5. Let G be the context free grammar G of Example 3b. Then $G' = \langle \Omega, \Sigma', P', Z' \rangle$ is a normal form grammar such that $L(G) = L(G')$ where

(i) $N_o' = \{Z, A, B, C\}$, $N_6' = \{F, G, H, J, K, L, M\}$;

(ii) $P' = \{Z \to MABCABC, M(x, y, z, u, v, w) \to M(F(x, y, z, u, v, w), G(x, \ldots, w), H(x, \ldots, w), J(x, \ldots, w), K(x, \ldots, w), L(x, \ldots, w)), M(x, \ldots, w) \to gxyz, F(x, \ldots, w) \to f(u, x), G(x, \ldots, w) \to f(v, y), H(x, \ldots, w) \to f(w, z), J(x, \ldots, w) \to u, K(x, \ldots, w) \to v, L(x, \ldots, w) \to w, A \to a, B \to b, C \to c\}$;

(iii) $Z' = Z$.

Theorem 6. Suppose L_1 and L_2 are (indexed families of) context free sets of term, $L \sqsubseteq (W_\Omega)_i$ is a context free set, and $a \in \Omega_{\langle \lambda,i \rangle}$. Then $L_1 \cup L_2$, $L_1 \cdot_a L$ and L^{*a} are all context free.

Proof: We will give only the construction of a grammar $G' = \langle \Omega, \Sigma', P', Z' \rangle$ for $L_1 \cdot_a L$ as the proofs are simple. Let $L_1 = L(G^1)$, $L = L(G)$ where $G^1 = \langle \Omega, \Sigma^1, P^1, Z^1 \rangle$, $G = \langle \Omega, \Sigma, P, Z \rangle$ are grammars for L_1 and L, respectively. Assume $N^1 \cap N = \{\phi\}$. Then

(i) $N' = N^1 \cup N^2$ less the axiom Z of G;

(ii) $P' = \{p \in P^1 | p$ is not of the form $A \to a\} \cup$ $\{p \in P | p$ is not of the form $Z \to t\} \cup \{A \to t | A \in N^1_{\langle \lambda, i \rangle}$, $A \to a$ is in P^1 and $t \in (W_\Sigma)_i$, $Z \to t$ is in $P\}$;

(iii) $Z' = Z^1$.

Thus G' has derivations which are the same as those of G^1 until an a would have appeared, at which point we have a derivation in G of a term in L. It is clear that $L(G') = L_1 \cdot_a L$.

We now set out to prove that the class of (indexed families of) context free sets of terms is closed under the operation of intersection with (indexed families of) recognizable sets. This result is a corollary of another important and interesting closure result: closure of the class under non-deterministic linear finite-state transformations. This is an idea due to Rounds and the reader is referred to Rounds (1), (2), (3) for full details.

A non-deterministic linear finite-state transformation (NLFST) is a four-tuple $T = \langle \Omega, Q, Q^{In}, \Pi \rangle$ where:

(i) Ω is a many-sorted alphabet;

(ii) $Q = \{Q_i\}_{i \in I}$ is a family of finite sets called states;

(iii) $Q^{In} \subsetneq Q$ is a choice of initial states;

(iv) Π is a finite set of productions of the form $(q, f(x_0, \ldots, x_{n-1})) \to t$ where $f \in \Omega_{\langle w,i \rangle}$, $q \in Q_i$, and $t \in W_\Omega(Q \times X_w)_i$. We require that each x_j, $0 \leq j \leq n-1$, appear at most once in t. We say such a production is linear.

We define a binary relation $\underset{T}{\Longrightarrow}$ called <u>direct derivation</u> on $W_\Omega(Q \times W_\Omega)$ as follows: $t' \underset{T}{\Longrightarrow} t''$ if and only if there exists $r \in \hat{W}_\Omega(Q \times W_\Omega, X_i)$, a production $(q, fx_o \ldots x_{n-1}) \to Sub_w(t; (q_o, x_o), \ldots, (q_{n-1}, x_{n-1}))$ in Π for $f \in \Omega_{\langle w,i \rangle}$ and $q \in Q_i$ such that $t' = Sub_i(r; (q, ft_o \ldots t_{n-1}))$ and $t'' = Sub_i(r; Sub_w(t; (q_o, t_o), \ldots, (q_{n-1}, t_{n-1})))$. Let the relation $\overset{*}{\underset{T}{\Longrightarrow}}$, called <u>derivation</u>, be the reflexive, transitive closure of the relation $\underset{T}{\Longrightarrow}$. Let $t \in (W_\Omega)_i$, $q \in Q_i$. Then:

$$\overline{T}(q, t) = \{t' \in W_\Omega(Q \times W_\Omega)_i \mid (q, t) \overset{*}{\underset{T}{\Longrightarrow}} t'\};$$

$$T(q, t) = \{t' \in W_\Omega \mid (q, t) \overset{*}{\underset{T}{\Longrightarrow}} t'\}.$$

Let $C \sqsubseteq W_\Omega$. Then $T(C) = \bigsqcup_{(q \in \hat{Q}^{In}, t \in \hat{C})} T(q, t)$.

We may ask at this point what relation, if any, NLFST have to semi-Thue grammars. It turns out that we may associate with each $T(q, t)$ a context sensitive (in fact, monotonic) grammar $G = \langle \Omega, \Sigma, P, Z \rangle$ such that $T(q, t) = L(G)$. This is done as follows: Let $N_{\langle i,i \rangle} = Q_i$ and $Z \in N_{\langle \lambda,j \rangle}$ for $t \in (W_\Omega)_j$. For each $(\overline{q}, f(x_o, \ldots, x_{n-1})) \to Sub_w(s; (q_o, x_o), \ldots, (q_{n-1}, x_{n-1}))$ in Π, let $\overline{q}(fx_o \ldots x_{n-1}) \to Sub_w(s; q_o(x_o), \ldots, q_{n-1}(x_{n-1}))$ be in P. Let $Z \to q(t)$ be in P. Then it should be clear that $Z \overset{*}{\underset{G}{\Longrightarrow}} t'$ if and only if $t' \in \overline{T}(q, t)$. Also $L(G) = T(q, t)$.

A production $F(x_o, \ldots, x_{n-1}) \to t$ in a context free term grammar $G = \langle \Omega, \Sigma, P, Z \rangle$ is said to be <u>useless</u> if there is no $t' \in \hat{W}_\Omega(X)$ such that $t \overset{*}{\underset{G}{\Longrightarrow}} t'$.

Lemma 7. If G' is obtained from G by discarding all uesless productions, then L(G') = L(G). (Note that there is an effective procedure for deciding whether a production in G is useless or not. See Corollary of Theorem VI.2.3).

Proof. It is clearly true that $L(G') \subseteq L(G)$. For the converse, we proceed, informally, as follows: Let $t \in L(G)$ and let $t_i \underset{G}{\Longrightarrow} t_{i+1}$ be the last application of a useless production p in G. Then, by the definition of derivation, there exists $r \in \hat{W}_\Sigma(X_j)$ such that $t_i = Sub_j(r; Fs_o \dots s_{n-1})$ and $t_{i+1} = Sub_j(r; Sub_w(\nu; s_o, \dots, s_{n-1}))$ for $p = F(x_o, \dots, x_{n-1}) \to \nu$ and $F \in N_{\langle w,j \rangle}$. But then there is no $r' \in \hat{W}_\Omega$ such that $Sub_j(r; Sub_w(\nu; s_o, \dots, s_{n-1})) \overset{*}{\underset{G}{\Longrightarrow}} r'$. Therefore we can assume that no further productions apply in $Sub_j(r; Sub_w(\nu; s_o, \dots, s_{n-1}))$ and if $t_{i+1} \underset{G}{\Longrightarrow} t_{i+2}$ via p' in P (of G), then $t_i \underset{G}{\Longrightarrow} t_{i+2}$ via p'. This eliminates the use of (the useless production) p.

Theorem 8. The class (indexed families of) context free sets of terms is closed under the operation of non-deterministic linear finite-state transformations.

Proof. We give here a sketch of the proof and the interested reader is referred to Rounds (1), (2) for the details (of the generalised case). Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a reduced form grammar from which all useless productions have been eliminated. Let $T = \langle \Omega, Q, Q^{In}, \Pi \rangle$ be a NLFST. The technique used in the proof is to allow T and G to 'run simultaneously', T transforming a symbol of Ω as soon as G has produced it. We obtain the system $G' = \langle \Omega, \langle Q,\Sigma \rangle, P', S \rangle$ as follows:

(i) If $F(x_o, \dots, x_{n-1}) \to fx_o \dots x_{n-1}$ is in P, put $(q, F(x_o,\dots,x_{n-1})) \to Sub_w(t; (q_o, x_o), \dots, (q_{n-1}, x_{n-1}))$ into P' for each production $(q, fx_o \dots x_{n-1}) \to Sub_w(t; (q_o, x_o), \dots, (q_{n-1}, x_{n-1}))$ in $\Pi(f \in \Omega_{\langle w,i \rangle})$;

(ii) If $F(x_o, \ldots, x_{n-1}) \rightarrow t$ is in $P(F \in N_{\langle w,i \rangle}, t \in W_\Sigma(X_w)_i)$, then put the production $(q, F(x_o, \ldots, x_{n-1})) \rightarrow (q, t)$ into P' for each $q \in Q_i$.

(iii) Let $S = \{(q, Z) | q \in Q_i^{In}, Z \in N_{\langle \lambda,i \rangle}\}$.

This system G' is the generalisation to many-sorted alphabets of what Rounds calls a <u>creative term grammar</u> (<u>dendrogrammar</u>). It can be shown that T(L(G)) = L(G'). Also with any creative term grammar G', we can associate a context free term grammar G" such that L(G") = L(G')(=T(L(G)). This then proves the theorem.

<u>Corollary</u>. The class of (indexed families of) context free sets of terms is closed under the operation of intersection with (indexed families of) recognizable sets of terms.

<u>Proof</u>. Every (indexed family of) recognizable set(s) is the domain of a NLFST which acts as the identity when defined.

<u>Theorem 9</u>. Suppose $\pi : \Omega \rightarrow \Delta$ is a projection and $G = \langle \Omega, \Sigma, P, Z \rangle$ and $G' = \langle \Delta, \Sigma', P', Z' \rangle$ are context free grammars over Ω and Δ, respectively. Then $\overline{\pi}(L(G))$ and $\overline{\pi}^{-1}(L(G'))$ are context free over Δ and Ω, respectively.

<u>Proof</u>. Similar to the proof for the analogous result for recognizable (indexed families of) sets.

V THE FUNDAMENTAL THEOREM

V.0. Introduction

This chapter contains the basic result of our work. It is a generalisation of such concepts as 'yield', 'leaf profile', 'frontier function', etc. (see Brainerd (1), Rounds (1), (2), (3); Thatcher (1), (2), Mezei and Wright). We call our result the 'Fundamental Theorem' because we believe it will play a role in Formal Language Theory similar to the Fundamental Theorem of Algebra (of Universal Algebra). The reasons for this belief will become clear in the next chapter.

V.1 Who Needs Context Free Sets?

Let $\Omega \sqsubseteq \Sigma$ be a many-sorted alphabet. (Note that from now on we will always speak of a context free set over some alphabet Ω in the context of $\Sigma \sqsupseteq \Omega$, the alphabet of non-terminals and terminals of the grammar which generates that set. Similarly, we will use the notation $D_{()}(\Omega)$ in this context to mean $D_{()}(\Sigma)$ less the nullaries corresponding to $N = \Sigma - \Omega$. So this alphabet $D_{()}(\Omega) \sqsubseteq D_{()}(\Sigma)$ and we 'preserve' the relationship. In general, of course, this alphabet $D_{()}(\Omega) \sqsubseteq D_{()}(\Sigma)$ will not be the same as the alphabet normally denoted by $D_{()}(\Omega)$. In fact, the second alphabet will not usually be a subset (in the extended sense) of $D_{()}(\Sigma)$).
Consider the three algebras W_Σ, $D_{()}(W_\Sigma)$, and $W_{D_{()}(\Sigma)}$. There exists a unique homomorphism

$$\text{YIELD}: W_{D_{()}(\Sigma)} \to D_{()}(W_\Sigma) \ .$$

Note that the set $(W_\Sigma)_i$, $i \in I$, is also an element of the family of sets $D_{()}(W_\Sigma)$ of sort $\langle \lambda, i \rangle$.

<u>Theorem 1.</u> Suppose $G = \langle \Omega, \Sigma, P, Z \rangle$ is a context free grammar and $L(G) \sqsubseteq W_\Omega$. Then we can effectively find a regular grammar

$G' = \langle D_{()}(\Omega),\ D_{()}(\Sigma),\ P',\ Z' \rangle$ such that $L(G')_{\langle w,i \rangle} = \phi$ for $w \neq \lambda$ (so $L(G')$ can be indexed by I) and $L(G) = \{YIELD_{\langle \lambda,i \rangle}(L(G'))\}_{i \varepsilon I}$ (that is, $L(G)_i = YIELD_{\langle \lambda,i \rangle}(L(G'))_{\langle \lambda,i \rangle}$.

Proof. Let G be in normal form. (We note here that the normal form theorem is not necessary for this proof but makes the proof simpler.) G' is defined as follows:

(i) For each $F \varepsilon N$ denote by F' the corresponding symbol in N'. If the type of F is $\langle w,i \rangle \varepsilon I^* \times I$, then the type of F' is $\langle \lambda, \langle w,i \rangle \rangle \ \varepsilon\ D_{()}(I)$. Essentially, we are just relabelling the non-terminals of G' for convenience;

(ii) P' is obtained from P as follows -

(a) For $F(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}) \rightarrow$ $G(H_o(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}), \ldots, H_{m-1}(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}))$ in P let $F' \rightarrow c_{\langle w,v,i \rangle} G' H_o' \ldots H'_{m-1}$ be in P' where F is of type $\langle w, i \rangle$, $w = w_o \ldots w_{n-1}$, G is of type $\langle v,i \rangle$, and each H_j, $0 \leq j \leq m-1$, is of type $\langle w, v_j \rangle$;

(b) For $F(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}) \rightarrow f x_{j_o,w_{j_o}} \ldots x_{j_k,w_{j_k}}$ in P let $F' \rightarrow c_{\langle w,v,i \rangle} f\ \delta_v^{j_o} \ldots \delta_v^{j_k}$ be in P' where F is of type $\langle w, i \rangle$, $w = w_o \ldots w_{n-1}$, f is of type $\langle v, i \rangle$, $v = w_{j_o} \ldots w_{j_k}$;

(c) For $F(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}) \rightarrow x_{j,w_j}$ in P let $F' \rightarrow \delta_w^{j+1}$ be in P' where F is of type $\langle w, w_j \rangle$, $w = w_o \ldots w_{n-1}$.

Obviously G' is a regular grammar (or, at least, easily reducible to one which is regular by a version of the Corollary to the Standard Form Theorem. Note that (ii)(b) may introduce terminal productions which are not strictly regular). We claim that, given $t_1 \varepsilon W_\Sigma(X_w)_i$ and $s_1 \varepsilon (W_{D_{()}(\Sigma)})_{\langle\lambda,i\rangle}$ such that $YIELD_{\langle\lambda,i\rangle}(s_1) = t_1$, $t_1 \overset{q}{\underset{G}{\Longrightarrow}} t_2$ in q steps via the productions $p_o, \ldots, p_{q-1}$ in P if and only if $s_1 \overset{q}{\underset{G'}{\Longrightarrow}} s_2$ via the corresponding productions $p_o', \ldots, p_{q-1}'$ of P' and $YIELD_{\langle\lambda,i\rangle}(s_2) = t_2$. If the claim is true, then it is easy to see that $Z \overset{*}{\underset{G}{\Longrightarrow}} t \varepsilon (W_\Omega)_i$ if and only if $Z' \overset{*}{\underset{G'}{\Longrightarrow}} t' \varepsilon (W_{D_{()}(\Omega)})_{\langle\lambda,i\rangle}$ and $YIELD_{\langle\lambda,i\rangle}(t') = t$. Thus $YIELD(L(G')) = L(G)$.

Proof of claim: The proof will be by induction on the length of the derivation (using only the outside-in mode allowed for normal form grammars).

length $q = 0$: obvious.

length $q > 0$: Suppose the claim is true for all derivations of length $< q$. Let $t_1 \overset{q}{\underset{G}{\Longrightarrow}} t_2$, $t_1, t_2 \varepsilon (W_\Sigma)_i$. Then $t_1 \overset{q-1}{\underset{G}{\Longrightarrow}} t_2' \underset{G}{\Longrightarrow} t_2$, the last step via the production p_{q-1} in P. There exists $s'_2 \varepsilon (W_{D_{()}(\Sigma)})_{\langle\lambda,i\rangle}$ such that $YIELD_{\langle\lambda,i\rangle}(s_2') = t_2'$ and $s_1 \overset{q-1}{\underset{G'}{\Longrightarrow}} s_2'$ (by the induction hypothesis).

Case 1: If $t_2' = H(\xi_o, \ldots, \xi_{m-1})$ then $s'_2 = Sub_{\langle w,i\rangle}(\tilde{s}_2', H')$ where $\tilde{s}_2' \varepsilon W_{D_{()}(\Sigma)}(X_{\langle w,i\rangle})_{\langle\lambda,i\rangle}$ (where $\langle w,i\rangle$ is a string of length one over the alphabet $I^* \times I$). Thus $YIELD_{\langle\lambda,i\rangle}(\tilde{s}_2') =$ $Sub_w(\phi(x_{o,\langle w,i\rangle}); \xi_o, \ldots, \xi_{m-1})$ where ϕ is an assignment to $x_{o,\langle w,i\rangle}$. Thus if $\phi(x_{o,\langle w,i\rangle}) = H(x_o, \ldots, x_{m-1})$ then the above becomes $H(\xi_o, \ldots, \xi_{m-1})$.

(a) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to G(H_o(x_o, \ldots, x_{m-1}), \ldots, H_{k-1}(x_o, \ldots, x_{m-1}))$ where $H \varepsilon \Sigma_{\langle w,i\rangle}$, then:

$$t_2 = G(H_o(\xi_o, \ldots, \xi_{m-1}), \ldots, H_{k-1}(\xi_o, \ldots, \xi_{m-1}));$$

$s_2 = \mathrm{Sub}_{\langle w,i\rangle}(\tilde{s}_2'; cG'H_o' \ldots H'_{k-1})$ for some $c \in D_{()}(\Sigma)$; and

$$\begin{aligned}\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) &= \mathrm{Sub}_w(\mathrm{YIELD}_{\langle w,i\rangle}(cG'H'_o \ldots H'_{k-1}); \xi_o, \ldots, \xi_{m-1})\\ &= G(H_o(\xi_o, \ldots, \xi_{m-1}), \ldots, H_{k-1}(\xi_o, \ldots, \xi_{m-1}))\\ &= t_2 .\end{aligned}$$

(b) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to fx_{j_o} \ldots x_{j_k}$, then:

$t_2 = f(\xi_{j_o}, \ldots, \xi_{j_k})$;

$s_2 = \mathrm{Sub}_{\langle w,i\rangle}(\tilde{s}_2'; cf\delta_w^{j_o+1} \ldots \delta_w^{j_k+1})$ where $H \in \Sigma_{\langle w,i\rangle}$, $c \in D_{()}(\Sigma)$; and

$$\begin{aligned}\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) &= \mathrm{Sub}_w(\mathrm{YIELD}_{\langle w,i\rangle}(cf\,\delta_w^{j_o+1} \ldots \delta_w^{j_k+1}); \xi_o, \ldots, \xi_{m-1})\\ &= f\xi_{j_o} \ldots \xi_{j_k}\\ &= t_2 .\end{aligned}$$

(c) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to x_j$, $0 \leq j \leq m-1$, then:

$t_2 = \xi_j$;

$s_2 = \mathrm{Sub}_{w_j}(\tilde{s}'_2; \delta_w^{j+1})$ where $H \in \Sigma_{\langle w,i\rangle}$;

and

$$\begin{aligned}\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) &= \mathrm{Sub}_{w_j}(\mathrm{YIELD}_{\langle w,w_j\rangle}(\delta_w^{j+1}); \xi_o, \ldots, \xi_{m-1})\\ &= \xi_j\\ &= t_2 .\end{aligned}$$

Case 2: If $t_2' = \mathrm{Sub}_i(r; H\xi_o \ldots \xi_{m-1})$, where $r \in \hat{W}(X_i)$, and p_{q-1} applies to H, then $s_2' = \mathrm{Sub}_{\langle i,i\rangle}(r'; cH'\xi_o' \ldots \xi'_{m-1})$ for some $c \in D_{()}(\Sigma)$ where $H \in \Sigma_{\langle w,i\rangle}$ and $\mathrm{YIELD}_{\langle\lambda,w_j\rangle}(\xi_j') = \xi_j$ for each $0 \leq j \leq m-1$;

$\mathrm{YIELD}_{\langle\lambda,i\rangle}(cH'\xi_o' \ldots \xi'_{m-1}) = H(\xi_o, \ldots, \xi_{m-1})$; and

$\mathrm{YIELD}_{\langle i,i\rangle}(r') = \mathrm{Sub}_i(r; \phi(x_{o,\langle i,i\rangle}))$ where ϕ is an assignment to $x_{o,\langle i,i\rangle}$.

(a) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to G(H_o(x_o, \ldots, x_{m-1}), \ldots, H_{k-1}(x_o, \ldots, x_{m-1}))$, where $H \in \Sigma_{\langle w,i\rangle}$, then:

$t_2 = \mathrm{Sub}_i(r;\ G(H_o(\xi_o, \ldots, \xi_{m-1}), \ldots, H_{k-1}(\xi_o, \ldots, \xi_{m-1})))$;

$s_2 = \mathrm{Sub}_{\langle i,i\rangle}(r';\ c(dG'H_o' \ldots H_{k-1}')\xi_o' \ldots \xi_{m-1}')$ for some c, $d \in D_{()}(\Sigma)$;

and

$$\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) = \mathrm{YIELD}_{\langle\lambda,i\rangle}(\mathrm{Sub}_{\langle i,i\rangle}(r';\ c(d(G'H_o' \ldots H_{k-1}')\xi_o' \ldots \xi_{m-1}')))$$

$$= \mathrm{Sub}_i(r;\ \mathrm{YIELD}_{\langle\lambda,i\rangle}(c(dG'H'_o \ldots H_{k-1}')\xi_o' \ldots \xi_{m-1}'))$$

$$= \mathrm{Sub}_i(r;\ G(H_o(\xi_o,\ldots,\xi_{m-1}),\ldots,H_{k-1}(\xi_o,\ldots,\xi_{m-1})))$$

$$= t_2 .$$

(b) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to fx_{j_o} \ldots x_{j_k}$, then:

$t_2 = \mathrm{Sub}_i(r;\ f\xi_{j_o} \ldots \xi_{j_k})$;

$s_2 = \mathrm{Sub}_{\langle i,i\rangle}(r';\ c(df\delta_w^{j_o+1} \ldots \delta_w^{j_k+1})\xi_o' \ldots \xi_{m-1}')$ for c, $d \in D_{()}(\Sigma)$ and $H \in \Sigma_{\langle w,i\rangle}$;

and

$$\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) = \mathrm{YIELD}_{\langle\lambda,i\rangle}(\mathrm{Sub}_{\langle i,i\rangle}(r'; c(df\delta_w^{j_o+1} \ldots \delta_w^{j_k+1})\xi_o' \ldots \xi_{m-1}'))$$

$$= \mathrm{Sub}_i(r; f\xi_{j_o} \ldots \xi_{j_k})$$

$$= t_2 .$$

(c) If p_{q-1} is $H(x_o, \ldots, x_{m-1}) \to x_j$, $0 \leq j \leq m-1$, $H \in \Sigma_{\langle w,i\rangle}$, then:

$t_2 = \mathrm{Sub}_i(r; \xi_j)$;

$s_2 = \mathrm{Sub}_{\langle i,i\rangle}(r'; c\delta_w^{j+1}\xi_o' \ldots \xi_{m-1}')$ for some $c \in D_{()}(\Sigma)$;

and

$$\mathrm{YIELD}_{\langle\lambda,i\rangle}(s_2) = \mathrm{YIELD}_{\langle\lambda,i\rangle}(\mathrm{Sub}_{\langle i,i\rangle}(r'; c\delta_w^{j+1}\xi_o' \ldots \xi_{m-1}'))$$

$$= \mathrm{Sub}_i(r; \mathrm{YIELD}_{\langle\lambda,w_j\rangle}(\xi_j'))$$

$$= \mathrm{Sub}_i(r; \xi_j)$$

$$= t_2 .$$

Theorem 2. Let $G = \langle D_{()}(\Omega), D_{()}(\Sigma), P, Z\rangle$ be a regular grammar such that $L(G)_{\langle w,i\rangle} = \phi$ if $w \neq \lambda$. (Thus we can consider $L(G)$ to be indexed by I.) Then we can effectively find a context free grammar $G' = \langle \Omega, \Sigma, P', Z'\rangle$ such that $\mathrm{YIELD}(L(G)) = L(G')$.

Proof. Let G be as given. We obtain G' as follows:

(i) For each $F \in N$, let $F' \in N'$ of type $\langle w, i\rangle$ where F is of type $\langle \lambda, \langle w,i\rangle\rangle$. We are again relabelling the non-terminals of G' for convenience;

(ii) For each production $F \to t$ in P, let $F'(x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}) \to \mathrm{YIELD}_{\langle w,i\rangle}(t)$ be in P' where F is of sort $\langle w,i\rangle$ and $\{x_{o,w_o}, \ldots, x_{n-1,w_{n-1}}\} = X_w$.

The rest of the proof is exactly analogous to that of the previous theorem and is omitted.

<u>Example 1</u>. Let $G' = \langle \Omega, \Sigma', P', Z' \rangle$ be the normal form grammar of Example IV.1.5. Then the regular grammar $G'' = \langle D_{()}(\Omega), D_{()}(\Sigma'), P'', Z'' \rangle$ is obtained, as in Theorem 1, where

(a) $D_{()}(\Sigma')$ is a many sorted alphabet sorted by $D_{()}(I) = \{0,1,2,3,6\}$ (we have again adopted the convention used in Example I.4.3) in which: $D_{()}(\Sigma')_{\langle \lambda,o \rangle} = \{a,b,c,Z'',A',B',C'\}$; $D_{()}(\Sigma')_{\langle \lambda,1 \rangle} = \{\delta^1_1\}$; $D_{()}(\Sigma')_{\langle \lambda,2 \rangle} = \{\delta^1_2, \delta^2_2, f\}$; $D_{()}(\Sigma')_{\langle \lambda,3 \rangle} = \{\delta^1_3, \delta^2_3, \delta^3_3, g\}$; $D_{()}(\Sigma')_{\langle \lambda,6 \rangle} = \{\delta^1_6, \ldots, \delta^6_6, F',G',H',J',K',L',M'\}$. $D_{()}(\Sigma')_{\langle mn\ldots n,n \rangle} = \{c_{\langle m,n,n \rangle}\}$ for m = 1,2,3 or 6 and n = 0,1,2,3 or 6.

(b) $D_{()}(\Omega)$ is $D_{()}(\Sigma')$ less the individual symbols Z'',A',B',C' of type $\langle \lambda, 0 \rangle$ and the individual symbols F',G',H',J',K',L',M' of type $\langle \lambda,6 \rangle$. Note that this is not the alphabet of Example I.4.3 but the alphabet indicated by the convention stated at the beginning of this chapter.

(c) P'' is enumerated below:

(1) $Z'' \to c_{\langle 6,o,o \rangle} M'A'B'C'A'B'C'$;

(2) $M' \to c_{\langle 6,6,6 \rangle} M'F'G'H'J'K'L'$;

(3) $M' \to c_{\langle 3,6,6 \rangle} g\delta^1_6\delta^2_6\delta^3_6$;

(4) $F' \to c_{\langle 2,6,6 \rangle} f\delta^4_6\delta^1_6$;

(5) $G' \to c_{\langle 2,6,6 \rangle} f\delta^5_6\delta^2_6$;

(6) $H' \to c_{\langle 2,6,6 \rangle} f\delta^6_6\delta^3_6$;

(7) $J' \rightarrow \delta_6^4$;

(8) $A' \rightarrow a$;

(9) $K' \rightarrow \delta_6^5$;

(10) $B' \rightarrow b$;

(11) $L' \rightarrow \delta_6^6$;

(12) $C' \rightarrow c$.

An example of a derivation is:

$$Z'' \underset{G''}{\Longrightarrow} c_{<6,o,o>} M'A'B'C'A'B'C' \quad \text{(by rule (1))}$$

$$\underset{G''}{\Longrightarrow} c_{<6,o,o>} c_{<3,6,6>} g\delta_3^1\delta_3^2\delta_3^3 A'B'C'A'B'C' \quad \text{(by rule (3))}$$

$$\overset{*}{\underset{G''}{\Longrightarrow}} c_{<6,o,o>} c_{<3,6,6>} g\delta_3^1\delta_3^2\delta_3^3 \text{ abcabc (by two applications each of rules (8), (10) and (12)).}$$

Consider $\text{YIELD}_{<\lambda,o>}(c_{<6,o,o>} c_{<3,6,6>} g\delta_3^1\delta_3^2\delta_3^3 \text{ abcabc})$. It is obviously gabc ε L(G"). Another example of a derivation is

$$Z'' \underset{G''}{\Longrightarrow} c_{<6,o,o>} M'A'B'C'A'B'C' \quad \text{(by rule (1))}$$

$$\underset{G''}{\Longrightarrow} c_{<6,o,o>} c_{<6,6,6>} M'F'G'H'J'K'L'A'B'C'A'B'C' \quad \text{(by rule (2))}$$

$$\underset{G''}{\Longrightarrow} c_{<6,o,o>} c_{<6,6,6>} c_{<6,6,6>} M'F'G'H'J'K'L'F'G'H'J'K'L'A'B'C'A'B'C' \quad \text{(by rule (2) again)}$$

$$\underset{G''}{\Longrightarrow} c_{<6,o,o>} c_{<6,6,6>} c_{<6,6,6>} c_{<3,6,6>} g\delta_3^1\delta_3^2\delta_3^3 F'G'H'J'K'L'F'G'H'J'K'L'A'B'C'A'B'C' \quad \text{(by rule (3))}$$

$$\overset{*}{\underset{G''}{\Longrightarrow}} c_{\langle 6,o,o\rangle} c_{\langle 6,6,6\rangle} c_{\langle 6,6,6\rangle} c_{\langle 3,6,6\rangle} g\delta_3^1\delta_3^2\delta_3^3$$

$$c_{\langle 2,6,6\rangle} f\delta_6^4\delta_6^1 \, c_{\langle 2,6,6\rangle} f\delta_6^5\delta_6^2 \, c_{\langle 2,6,6\rangle} f\delta_6^6\delta_6^3$$

$$\delta_6^4\delta_6^5\delta_6^6 \; F'G'H'J'K'L'A'B'C'A'B'C' \quad \text{(by rules (4), (5), (6), (7), (9), (11))}$$

$$\overset{*}{\underset{G''}{\Longrightarrow}} c_{\langle 6,o,o\rangle} c_{\langle 6,6,6\rangle} c_{\langle 6,6,6\rangle} c_{\langle 3,6,6\rangle} g\delta_3^1\delta_3^2\delta_3^3$$

$$c_{\langle 2,6,6\rangle} f\delta_6^4\delta_6^1 \, c_{\langle 2,6,6\rangle} f\delta_6^5\delta_6^2 \, c_{\langle 2,6,6\rangle} f\delta_6^6\delta_6^3\delta_6^4\delta_6^5\delta_6^6$$

$$c_{\langle 2,6,6\rangle} f\delta_6^4\delta_6^1 \, c_{\langle 2,6,6\rangle} f\delta_6^5\delta_6^2 \, c_{\langle 2,6,6\rangle} f\delta_6^6\delta_6^3\delta_6^4\delta_6^5\delta_6^6 \; abcabc$$

(by rules (4), (5), (6), (7), (9), (11), (8), (10), (12)).

Call the result of the above derivation t. Then $\text{YIELD}_{\langle\lambda,o\rangle}(t)$ is seen to be gfafaafbfbbfcfcc ε $L(G'')$.

Example 2. Let $G' = \langle\Omega, \Sigma', P', Z'\rangle$ be the normal form grammar of Example IV.1.4. Then the regular grammar $G'' = \langle D_{()}(\Omega), D_{()}(\Sigma'), P'', Z''\rangle$ is obtained, as in Theorem 1, where:

(a) $D_{()}(\Sigma')$ is a many-sorted alphabet sorted by $D_{()}(I) = \{\langle\lambda,o\rangle, \langle\lambda,1\rangle, \langle 1,o\rangle, \langle 1,1\rangle, \langle 10,0\rangle, \langle 10,1\rangle, \langle 11,0\rangle, \langle 11,1\rangle\}$ in which: $D_{()}(\Sigma')_{\langle\lambda,\langle\lambda,0\rangle\rangle} = \{\lambda, Z'', L'\}$; $D_{()}(\Sigma')_{\langle\lambda,\langle\lambda,1\rangle\rangle} = \{a, A', C'\}$; $D_{()}(\Sigma')_{\langle\lambda,\langle 1,0\rangle\rangle} = \phi$; $D_{()}(\Sigma')_{\langle\lambda,\langle 1,1\rangle\rangle} = \{\delta^1_{(1)}, B', D'\}$; $D_{()}(\Sigma')_{\langle\lambda,\langle 10,0\rangle\rangle} = \{\delta^2_{(10)}, *, S'\}$; $D_{()}(\Sigma')_{\langle\lambda,\langle 10,1\rangle\rangle} = \{\delta^1_{(10)}\}$; $D_{()}(\Sigma')_{\langle\lambda,\langle 11,0\rangle\rangle} = \phi$; $D_{()}(\Sigma')_{\langle\lambda,\langle 11,1\rangle\rangle} = \{\delta^1_{(11)}, \delta^2_{(11)}, +\}$. $D_{()}(\Sigma')_{\langle\langle w,i\rangle\langle v,w_o\rangle \ldots \langle v,w_{n-1}\rangle, \langle v,i\rangle\rangle} = \{c_{\langle w,v,i\rangle}\}$ for each $(w,v,i) \in \{1,10,11\} \times \{\lambda, 1, 10, 11\} \times I$.

(b) $D_{()}(\Omega)$, according to our convention, is $D_{()}(\Sigma')$ less the individual symbols Z", L' of type $\langle \lambda, \langle \lambda, 0\rangle\rangle$; the individual symbols A', C' of type $\langle \lambda, \langle \lambda, 1\rangle\rangle$; the individual symbols B', D' of type $\langle \lambda, \langle 1, 1\rangle\rangle$; and the individual symbol S' of type $\langle \lambda, \langle 10, 0\rangle\rangle$.

(c) P" is enumerated below:

(1) $Z'' \rightarrow c_{\langle 10,\lambda,0\rangle} S'A'L'$;

(2) $Z'' \rightarrow c_{\langle 10,\lambda,0\rangle} S'C'L'$;

(3) $C' \rightarrow c_{\langle 1,\lambda,1\rangle} B'A'$;

(4) $S' \rightarrow c_{\langle 10,10,0\rangle} {*}\delta^1{}_{10} \ \delta^2{}_{10}$;

(5) $B' \rightarrow c_{\langle 1,1,1\rangle} B'D'$;

(6) $B' \rightarrow c_{\langle 11,1,1\rangle} + \delta^1{}_1 \ \delta^1{}_1$;

(7) $D' \rightarrow c_{\langle 11,1,1\rangle} + \delta^1{}_1 \ \delta^1{}_1$;

(8) $A' \rightarrow a$;

(9) $L' \rightarrow \lambda$.

An example of a derivation is:

$$Z'' \underset{G''}{\Longrightarrow} c_{\langle 10,\lambda,0\rangle} S'A'L' \quad \text{(by rule (1))}$$

$$\underset{G''}{\Longrightarrow} c_{\langle 10,\lambda,0\rangle} c_{\langle 10,10,0\rangle} * \delta^1{}_{10} \ \delta^2{}_{10} \ A'L' \quad \text{(by rule (4))}$$

$$\overset{*}{\underset{G''}{\Longrightarrow}} c_{\langle 10,\lambda,0\rangle} c_{\langle 10,10,0\rangle} * \delta^1{}_{10} \ \delta^2{}_{10} \ a\lambda \quad \text{(by rules (8) and (9))} .$$

It is obvious that $\text{YIELD}_{\langle \lambda,0\rangle} (c_{\langle 10,\lambda,0\rangle} c_{\langle 10,10,0\rangle} * \delta^1{}_{10} \ \delta^2{}_{10} \ a\lambda) = {*}a\lambda$. An example of a longer derivation is:

$$Z'' \underset{G''}{\Longrightarrow} c_{<10,\lambda,0>} S'C'L' \quad \text{(by rule (2))}$$

$$\underset{G''}{\Longrightarrow} c_{<10,\lambda,0>} c_{<10,10,0>} * \delta^1_{10} \; \delta^2_{10} \; C'L' \quad \text{(by rule (4))}$$

$$\underset{G''}{\Longrightarrow} c_{<10,\lambda,0>} c_{<10,10,0>} * \delta^1_{10} \; \delta^2_{10} \; c_{<1,\lambda,1>} B'A'L' \quad \text{(by rule (3))}$$

$$\underset{G''}{\Longrightarrow} c_{<10,\lambda,0>} c_{<10,10,0>} * \delta^1_{10} \; \delta^2_{10} \; c_{<1,\lambda,1>} \, c_{<11,1,1>}$$

$$+ \; \delta^1_1 \; \delta^1_1 \; A'L' \quad \text{(by rule (6))}$$

$$\overset{*}{\underset{G''}{\Longrightarrow}} c_{<10,\lambda,0>} c_{<10,10,0>} * \delta^1_{10} \; \delta^2_{10} \; c_{<1,\lambda,1>} \, c_{<11,1,1>}$$

$$+ \; \delta^1_1 \; \delta^1_1 \; a\lambda \quad \text{(by rules (8) and (9))}.$$

Then if the result of the above derivation is t, $\mathrm{YIELD}_{<\lambda,0>}(t) =$ $*{+}aa\lambda \in L(G'')$.

The reader may already have noted that Theorem 2 is not a full converse of Theorem 1. This is because we have not yet assigned any meaning to the images under $\mathrm{YIELD}_{<w,i>}$ of recognizable sets of sort $<w,i>$ where $w \neq \lambda$. We do know that if U is a recognizable set of sort $<w,i>$ over $D_{()}(\Omega)$, then it is the pre-image under $\mathrm{YIELD}_{<w,i>}$ of derived operations of type $<w,i>$ over the algebra W_Ω. Consider the many-sorted (augmented) alphabet $\Omega(X_w)$ where $\Omega(X_w)_{<v,i>} = \Omega_{<v,i>}$ for $v \neq \lambda$ and $\Omega(X_w)_{<\lambda,i>} = \Omega_{<\lambda,i>} \sqcup (X_w)_i$. $\mathrm{YIELD}_{<w,i>}(U)$ can now be considered a context free set of sort i over the augmented alphabet $\Omega(X_w)$. Context free sets over $\Omega(X_w)$ can also be defined in the normal way using context free term grammars. We call such sets (defined by either method above) context free sets of derive operations of type $<w,i>$ over Ω. This definition includes the cases considered in Theorems 1 and 2 as simpler examples (namely, when $w = \lambda$).

We must make one more slight extension of our previous theory: Indexed families of recognizable sets of terms over Ω are indexed by I.

Over $D_{()}(\Omega)$, they are indexed by $D_{()}(I)$. Thus the image of $U \sqsubseteq W_{D_{()}(\Omega)}$ under YIELD is a family of context free sets (of derived operations) indexed also by $D_{()}(I)$. From now on, we will always mean this indexing set when we mention a family of context free sets over Ω. (The preceding discussion of families of context free sets in Chapter IV is now the special case where the sets indexed by $\langle w,i \rangle \in D_{()}(I)$ are empty for $w \neq \lambda$.)

As a consequence of this discussion and the two preceding theorems, we are ready to state the most important result in this work:

Fundamental Theorem: Let $\Omega \sqsubseteq \Sigma$ be a many-sorted alphabet. Given L, an indexed family of context free sets of terms over Ω, we can effectively find L', an indexed family of recognizable sets of terms over $D_{()}(\Omega)$ such that YIELD(L') = L and conversely.

VI. APPLICATIONS OF THE FUNDAMENTAL THEOREM

VI.0 Introduction

As the title suggests, this chapter is devoted to the study of the consequences of the Fundamental Theorem stated at the end of Chapter V. We will show that the results of conventional and generalised theories of formal languages are special cases of our more general theory. We will also extend the theories in light of our definition of derived operations.

We begin by providing a full generalisation of the operations of complex product and Kleene-closure (on sets of strings). (The definitions we presented in Chapter II are 'partial' in a sense to be explained below.) The classical Substitution Theorem is then generalised. The concepts of 'yield', 'leaf profile', etc., are extended and clarified. We then proceed to discuss the operation of homomorphism (with respect to composition) and present an algebraic definition of regular, context free and indexed sets of terms ('indexed' not being here used in the sense we have so far meant it).

VI.1. Substitution

Let Σ be a many-sorted alphabet. Recall the definitions of a-complex product and a-Kleene closure for $a \in \Sigma_{\langle\lambda,i\rangle}$. We subsume these definitions in the following generalisation:

(i) Suppose $f \in \Sigma_{\langle w,i\rangle}$, $t \in \hat{W}_\Sigma(X_v)$, and $t' \in W_\Sigma(X_v, X_w)_i$. Define the operation $t \cdot_f t'$ of <u>f-substitution</u> as follows: $t \cdot_f t' = \text{YIELD}_{\langle\lambda,i\rangle}(\text{Sub}_{\langle w,i\rangle}(r; s'))$ where $t = \text{YIELD}_{\langle\lambda,i\rangle}(\text{Sub}_{\langle w,i\rangle}(r; f))$, $r \in \hat{W}_{D_{()}(\Sigma(X_v))}(X_{\langle w,i\rangle})$ and r has no occurrences of f, and $s' \in (W_{D_{()}(\Sigma(X_v))})_{\langle\lambda,\langle w,i\rangle\rangle}$ such that $\text{YIELD}_{\langle w,i\rangle}(s') = t'$. Informally we have just substituted the derived operation t' of type $\langle w,i\rangle$ for each occurrence of $f \in \Sigma_{\langle w,i\rangle}$ (taking care that the terms appearing in the argument list of an occurrence of f in t are substituted in the 'proper' place in t').

(ii) Given $U \sqsubseteq \hat{W}_\Sigma(X_v)$, $V \sqsubseteq W_\Sigma(X_v, X_w)_i$ and f as above, define the operation $U \cdot_f V$ of <u>f-complex product</u> as follows: $U \cdot_f V = \{t \mid YIELD_{\langle \lambda,i \rangle}$ $(Sub_u(r; s_o, \ldots, s_{n-1})) = t$ for some $i \in I$; $YIELD_{\langle\lambda,i\rangle}$ $(Sub_u(r; \underbrace{f, \ldots, f}_{n\ times})) \in U$ with exactly n occurrences of f; $s_j \in$ $(W_{D_{()}}(\Sigma(X_v)))_{\langle\lambda,\langle w,i\rangle\rangle}$ such that $YIELD_{\langle w,i\rangle}(s_j) \in V$ for $0 \leq j \leq n-1$; and $r \in W_{D_{()}}(\Sigma(X_v))(X_u)_{\langle\lambda,i\rangle}$, with $u = \langle w,i\rangle\langle w,i\rangle \ldots \langle w,i\rangle$ (a string of n $\langle w,i\rangle$'s), in which f does not occur$\}$. Informally, we obtain $U \cdot_f V$ by substituting some (not necessarily the same) element of V for each occurrence of the symbol f in an element of U. We again have $U \cdot_f \{f\} = U$ and $U \cdot_f \phi$ is the subset of U with no occurrence of f in any of its elements. We can, of course, extend this operation to an indexed family of sets $U \sqsubseteq W_\Sigma(X_v)$.

(iii) Given $U \sqsubseteq W_\Sigma(X_v, X_w)_i$ and f as above, define the operation U^{*f} of <u>f-Kleene closure</u> as follows: Let $V^o = \{fx_{o,w_o} \ldots x_{n-1,w_{n-1}}\}$ and $V^{m+1} = V^m \sqcup U \cdot_f V^m$. Then $U^{*f} = \bigsqcup_{m \in \underline{N}} V^m$.

We state without proof the important <u>Theorem 1 (Substitution)</u>: Suppose that L is an indexed family of context free sets of terms over Σ and that to each $f \in \Sigma$ there corresponds a context free set of terms L_f of the same type as f. Then if we (simultaneously) substitute each L_f ($f \in \Sigma$) for f in L, the resulting indexed family of sets of terms L' is context free.

VI.2. <u>'Yield' Theorems</u>

It is a well known result of the generalised theory that the class of context free sets of strings (which do not contain the empty string λ) over some alphabet Σ is equivalent to the class of sets of strings which are the 'yields' ('leaf profiles', 'frontiers') of recognizable sets over some non-sorted alphabet Δ which contains Σ as individual symbols. (See Brainerd (1), Mezei and Wright, Thatcher (1), (2), (3) and Rounds (2).) We note the anomaly of considering only the class of sets without the empty string. The following (really a corollary of the Fundamental Theorem) rectifies this situation:

Theorem 1. Let Ω be a finite set and $L \subseteq \Omega^*$ a context free set. Then $D_{()}(T(\Sigma))$ is a 2-sorted alphabet (sorted by $\{0, 1\}$) and there exists a recognizable set $U \subseteq (W_{D_{()}(T(\Omega))})_0$ such that $\text{YIELD}_0(U) = \phi_\Omega(L)$ (see Remark at the end of Chapter II for the definitions of $T(\Omega)$ and ϕ_Ω).

Corollary. Suppose $L \subseteq \Omega^+$ (L does not contain the empty string). Then there exists $U \subseteq (W_{D_{()}(T(\Omega))})_1$ such that $\overline{\phi}_\Omega(\text{YIELD}_1(U)) = L$ where $\overline{\phi}_\Omega : W_\Omega(\{x\}) \to \Omega^+$ is defined by $\overline{\phi}_\Omega(wx) = w$ for $w \in \Omega^+$.

The converse result will not be stated as it is any easy consequence of the Fundamental Theorem.

Example 1. Let $\Omega = \{a, b\}$ and consider the context free (string) grammar $G' = \langle \Omega, \Delta', P', Z' \rangle$ where:

(i) $N' = \{Z', A', B', C', D'\}$; and

(ii) $P' = \{Z' \to A'B', Z' \to A'C', C' \to D'B', D' \to A'C', D' \to A'B', A' \to a, B' \to b\}$.

$L(G') = \{a^n b^n | n > 0\}$. Consider the regular term grammar $G = \langle \Sigma, \Delta, P, Z \rangle$ of the second part of Example III.2.1. Σ is $D_{()}(T(\Omega))$ less the two individual symbols $\delta^1_{(0)}$ and $\delta^1_{(1)}$. It is clear that in the case of unary alphabets such as $T(\Omega)$, the projections $\delta^1_{(0)}$ and $\delta^1_{(1)}$ are 'redundant' and so can be dropped. We will follow this procedure. Then $\phi_\Omega(L(G')) = \text{YIELD}_1(L(G)_1)$. If we add the production $Z' \to \lambda$ to P' in G', then $L(G') = \{a^n b^n | n \geq 0\}$ and $\phi_\Omega(L(G')) = \text{YIELD}_0(L(G)_0)$.

We now go on to generalize a similar 'yield theorem' for context free sets of terms (see Rounds (2)). But first we must introduce two further definitions of string grammars:

An indexed grammar G is a 5-tuple $\langle \Omega, \Sigma, P, F, Z \rangle$ where:

(i) $N = \Sigma - \Omega$ is a set of non-terminal symbols;

(ii) Ω is a finite set of terminal symbols;

(iii) P is a finite set of productions of the form $A \to \alpha$ where $A \in N$ and $\alpha \in (NF^* \bigcup \Omega)^*$;

(iv) F is a finite set of flags where a flag $f \in F$ is a finite set of ordered pairs, or index productions, each of which is of the form $B \to \beta$ where $B \in N$ and $\beta \in (N \bigcup \Omega)^* = \Sigma^*$;

(v) Z is the axiom.

There are two rules of derivation: The string β is a consequence of (is directly generated by) the string α, $\alpha \underset{G}{\Longrightarrow} \beta$, if either of the following conditions holds:

(i) $\alpha = \alpha_1 A \xi \alpha_2$ for $\alpha_1, \alpha_2 \in (NF^* \bigcup \Omega)^*$, $A \in N$, $\xi \in F^*$;

$\beta = \alpha_1 X_1 \theta_1 X_2 \theta_2 \ldots X_k \theta_k \alpha_2$ for $X_1, \ldots, X_k \in \Sigma$; $A \to X_1 \eta_1 \ldots X_k \eta_k$ is a production in P, $\eta_1, \ldots, \eta_k \in F^*$ and for $1 \leq i \leq k$,

$$\theta_i = \begin{cases} \eta_i \xi \text{ if } X_i \in N \\ \lambda \text{ if } X_i \in \Omega \text{ (in which case } \eta_i = \lambda). \end{cases}$$

(ii) $\alpha = \alpha_1 B f \xi a_2$ for $\alpha_1, \alpha_2 \in (NF^* \bigcup \Omega)^*$, $B \in N$, $f \in F$, $\xi \in F^*$;

$\beta = \alpha_1 X_1 \theta_1 \ldots X_k \theta_k \alpha_2$ for $X_1, \ldots, X_k \in \Sigma$;

$B \to X_1 \ldots X_k$ is in f;

and for $1 \leq i \leq k$

$$\theta_i = \begin{cases} \xi \text{ if } X_i \in N \\ \lambda \text{ if } X_i \in \Omega. \end{cases}$$

In (i) we may informally say that A and its associated string of flags is replaced by the string obtained from the right hand side of a production for A by inserting a copy of ξ following each non-terminal symbol and its associated flags. In (ii), we say that the flag f is consumed by the non-terminal B.

Let $\overset{*}{\underset{G}{\Longrightarrow}}$ be the reflexive, transitive closure of $\underset{G}{\Longrightarrow}$ and L(G) the indexed set of strings generated by the indexed grammar G.

We again have a normal form for these grammars which we state in:

Lemma 2. Given an indexed grammar $G = \langle \Omega, \Sigma, P, F, Z \rangle$, we can effectively find another indexed grammar $G' = \langle \Omega, \Sigma', P', F', Z' \rangle$, said to be in normal form, such that L(G) = L(G'), where:

(i) Each production in P' is of the form:

(a) $A \to BC$ where $A \in N'$ and $B, C \in N' - \{Z'\}$;

(b) $A \to Bf$ where $A \in N'$, $B \in N' - \{Z'\}$, $f \in F'$;

(c) $A \to a$ where $A \in N'$ and $a \in \Omega$;

(d) $Z' \to \lambda$ if and only if L(G) contains the empty string;

(ii) Each index production is of the form $A \to B$ for $A \in N'$, $B \in N' - \{Z'\}$ and, for each flag $f \in F'$ and each $A \in N'$, f contains exactly one index production with A as left part.

Proof. A consequence of Theorem 4.5 of Aho (1) and Lemma 4.2.3 of Fischer (1).

A macro-grammar structure G is a 5-tuple $\langle \Omega, N, V, P, Z \rangle$ where:

(i) Ω is a finite set of terminal symbols;

(ii) $N = \bigsqcup_{n \in \underline{N}} N_n$ is a finite set of non-terminal symbols;

(iii) V is a finite set of variable symbols;

(iv) P is a finite set of productions of the form $F(x_o, \ldots, x_{n-1}) \to \nu$ where $F \in N_n$, $x_o, \ldots, x_{n-1}$ are distinct elements of V and ν is a term (see below) over $\Omega \bigcup \{x_o, \ldots, x_{n-1}\}$, N;

(v) Z is the axiom.

The set of terms over $\Omega \bigcup X$, N($X \sqsubseteq V$) is the least set of strings over $\Omega \bigcup X \bigcup N \bigcup \{(,), ", "\}$ satisfying:

(i) λ is a term;

(ii) a is a term if $a \in \Omega \bigcup X$;

(iii) $\sigma\tau$ is a term if σ and τ are;

(iv) $F(\sigma_o, \ldots, \sigma_{n-1})$ is a term if $F \in N_n$ and $\sigma_o, \ldots, \sigma_{n-1}$ are terms.

τ is said to be a subterm of σ if τ is a substring of σ and τ is a term. A subterm τ of a term σ is said to be at top level in σ if τ soes not appear in the argument list of any non-terminal in σ.

Given terms σ,τ over $\Omega \bigcup X$, N and a macro-grammar $G = \langle \Omega, N, V, P, Z \rangle$, τ is a consequence of (is directly generated by) σ by an unrestricted step, denoted $\sigma \underset{U,G}{\Longrightarrow} \tau$, in case the following condition holds: σ contains a subterm of the form $F(\xi_o, \ldots, \xi_{n-1})$ for $F \in N_n$ and $\xi_o, \ldots, \xi_{n-1}$ terms over $\Omega \bigcup X$, N; P contains a rule of the form $F(x_o, \ldots, x_{n-1}) \to \nu$ and τ results from σ by replacing a single occurrence of $F(\xi_o, \ldots, \xi_{n-1})$ by ν' where ν' is obtained from ν by replacing each occurrence of x_i by ξ_i for $0 \leq i \leq n-1$. σ directly generates τ by an outside-in step, denoted $\sigma \underset{OI,G}{\Longrightarrow} \tau$, if $\sigma \underset{U,G}{\Longrightarrow} \tau$ and the subterm of σ which is rewritten occurs at the top level of σ. Let $\overset{*}{\underset{U,G}{\Longrightarrow}}$ and $\overset{*}{\underset{OI,G}{\Longrightarrow}}$ be the reflexive, transitive closure of $\underset{U,G}{\Longrightarrow}$, $\underset{OI,G}{\Longrightarrow}$

respectively. Let L(G) be the set of strings generated by G.

We again have a normal form for these grammars which we state in:

Lemma 3. Given a macro-grammar structure $G = \langle \Omega, N, V, P, Z \rangle$ and that L(G) is generated by derivations in the unrestricted mode, then one can effectively find a macro-grammar structure $G' = \langle \Omega, N', V', P', Z' \rangle$ such that L(G) = L(G') and L(G') is generated by derivations in the outside-in mode, where the productions in P' are of the form:

(i) $F(x_o, \ldots, x_{n-1}) \rightarrow G(H_o(x_o,\ldots,x_{n-1}),\ldots,H_{m-1}(x_o,\ldots,x_{n-1}))$
for $F, H_o, \ldots, H_{m-1} \in N_n$ and $G \in N_m$; or

(ii) $F(x_o, \ldots, x_{n-1}) \rightarrow \nu$ where $\nu \in (\Omega \cup \{x_o, \ldots, x_{n-1}\})^*$
for $n \geq 0$.

Proof. See Fischer (1).

Theorem 4. Given an indexed grammar G, one can effectively find an outside-in macro-grammar structure G' such that L(G) = L(G') and conversely.

Proof. See Fischer (1).

We are now ready to state our promised 'yield theorem':

Theorem 5. Given an indexed grammar $G = \langle \Omega, \Sigma, P, F, Z \rangle$, we can effectively find a context free term grammar $G' = \langle D_{()}(T(\Omega)), \Sigma', P', Z' \rangle$ (where $T(\Omega)$ is the unary non-sorted alphabet corresponding to the string alphabet Ω) such that $\phi_\Omega(L(G)) = YIELD_o(L(G'))$.

Proof. We assume that G is in normal form. The proof is similar to the proof of the equivalence of the class of indexed languages and the class of languages generated by OI macro-grammar structures. Assume that $A_o, \ldots, A_n$ is an enumeration of N. For each $f \in F$ let $C_{f,o}, \ldots, C_{f,n}$

be the non-terminals in N such that $f = \{A_o \rightarrow C_{f,o}, \ldots, A_n \rightarrow C_{f,n}\}$.
Then we obtain G' as follows:

(i) $N'_{<\lambda,0>} = \{Z\}$, $N'_{<\lambda,1>} = \{A' | A \in N - \{Z\}\}$ and $N'_{<\underbrace{11\ldots1}_{n+1 \text{ times}},1>} = \{A'' | A \in N\}$;

(ii) P' is obtained as follows:

(a) If $Z \rightarrow BC$ is in P and Z, B, C $\in$ N, then let $Z' \rightarrow *+B'C'\lambda$ and $Z''(x_o, \ldots, x_n) \rightarrow + B''(x_o, \ldots, x_n)C''(x_o, \ldots, x_n)$ be in P', for $\{x_o, \ldots, x_n\} = X_w$, w is a string of n+1 1's;

(b) If $A \rightarrow BC$ is in P, $A \neq Z$, and A, B, C $\in$ N, then let $A' \rightarrow +B'C'$ and $A''(x_o, \ldots, x_n) \rightarrow +B''(x_o, \ldots, x_n)C''(x_o, \ldots, x_n)$ be in P' for X_w as in (a);

(c) If $Z \rightarrow Bf$ is in P, Z, B $\in$ N, then let $Z' \rightarrow * B''(C'_{f,0}, \ldots, C'_{f,n})$ and $A''(x_o, \ldots, x_n) \rightarrow B''(C''_{f,0}(x_o, \ldots, x_n), \ldots, C''_{f,n}(x_o, \ldots, x_n))$ be in P' for X_w as in (a);

(d) If $A \rightarrow Bf$ is in P, $A \neq Z$ and A, B $\in$ N, then let $A' \rightarrow B''(C'_{f,o}, \ldots, C'_{f,n})$ and $A''(x_o, \ldots, x_n) \rightarrow B''(C''_{f,o}(x_o, \ldots, x_n), \ldots, C''_{f,n}(x_o, \ldots, x_n))$ be in P' for X_w as in (a);

(e) If $Z \rightarrow a$ is in P for Z $\in$ N, a $\in \Omega$, then let $Z' \rightarrow *a\lambda$ and $Z''(x_o, \ldots, x_n) \rightarrow a$ be in P' for X_w as in (a);

(f) If $A \rightarrow a$ is in P, for $A \neq Z$ and A $\in$ N, a $\in \Omega$, then let $A' \rightarrow a$ and $A''(x_o, \ldots, x_n) \rightarrow a$ be in P' for X_w as in (a);

(g) If $Z \rightarrow \lambda$ is in P, then let $Z' \rightarrow \lambda$ be in P';

(h) Let $A_i''(x_o, \ldots, x_{n-1}) \rightarrow x_i$ be in P' for $0 \leq i \leq n$ (where $A_o, \ldots, A_n$ is the enumeration of N we have been considering).

We define the <u>representation map</u> ψ:

$(NF^* \bigsqcup \Omega)^* \rightarrow (W_{\Sigma'})_o$ inductively:

(i) $\psi(\lambda) = \lambda$;

(ii) $\psi(w) = {*}\theta(w)\lambda$ for $w \in (NF^* \bigsqcup \Omega)^+$ where $\theta : (NF^* \bigsqcup \Omega)^+ \rightarrow (W_{\Sigma'})_1$

is defined inductively by -

(a) $\theta(a) = a$ if $a \in \Omega$;

(b) $\theta(A) = A'$ if $A \in N$;

(c) $\theta(Af\eta) = A''(\theta(C_{f,o}\eta), \ldots, \theta(C_{f,n}\eta))$ if $Af\eta \in NF^+$ and $f \in F$, $\eta \in F^*$;

(d) $\theta(\alpha\beta) = {+}\theta(\alpha)\theta(\beta)$ if $\beta \in (NF^* \bigsqcup \Omega)^+$ and $\alpha \in NF^*$ or $\alpha \in \Omega$.

This clearly and uniquely defines $\psi(\theta)$ and $\psi(\theta)$ is clearly recursive.

There is a correspondence between productions in P and pairs of productions in P' obtained by rules (a) to (g) (except if $Z \rightarrow \lambda$ is in P, there corresponds to it only the production $Z' \rightarrow \lambda$ in P'). It is easy to verify that if we take left-most derivations in G and outside-in, leftmost derivations in G' (that is, choose the leftmost occurrence of a top level non-terminal where the outside-in mode leaves a choice), then for any $\alpha, \beta \in (NF^* \bigsqcup \Omega)^*$:

(i) $\alpha \underset{G}{\Longrightarrow} \beta$ via a production in P if and only if $\psi(\alpha) \underset{G'}{\Longrightarrow} \psi(\beta)$ via one of the two corresponding productions in P' obtained by rules (a) - (g);

(ii) $\alpha \underset{G}{\Longrightarrow} \beta$ by consuming a flag with the non-terminal symbol A_i if and only if $\psi(\alpha) \underset{G'}{\Longrightarrow} \psi(\beta)$ via the production $A_i''(x_o, \ldots, x_n) \rightarrow x_i$ in P' obtained by rule (h).

Hence for each $w \in \Omega^*$, $Z \overset{*}{\underset{G}{\Longrightarrow}} w$ if and only if $\psi(Z) = Z' \overset{*}{\underset{G'}{\Longrightarrow}} \psi(w)$. Thus we have our result: $\phi_\Omega(L(G)) = YIELD_o(L(G'))$.

Corollary: Given a macro-grammar G over the alphabet Ω one can effectively find a context free term grammar G' over $D_{()}(T(\Omega))$ such that $\phi_\Omega(L(G)) = YIELD_o(L(G'))$.

Proof. Consequence of the theorem and Theorem 4.

We may again note here that the 'yield theorems' previously published are results concerning indexed sets (over some alphabet Ω) not containing the empty string which relate them to context free term sets of sort 1 over the alphabet $D_{()}(T'(\Omega))$. The converse of the above theorem is obtained from:

Theorem 6. Given a set Ω and a context free term grammar $G = \langle D_{()}(T(\Omega)), \Sigma, P, Z \rangle$, we can effectively find a macro-grammar $G' = \langle \Omega, N', V', P', Z' \rangle$ such that $\phi_\Omega(L(G')) = YIELD_o(L(G))$.

Proof. Assume G is in normal form. G' is obtained as follows:

(i) $N_n' = \{A' | A \in N_{\langle w,i \rangle}$ and rank of A is n$\}$ for each $n \in \underline{N}$;

(ii) $V' = \{x_o, \ldots, x_{k-1}\}$ where k is the highest rank of any $A \in N$;

(iii) P' is obtained as follows:

(a) If $A(y_o, \ldots, y_{n-1}) \to B(C_o(y_o, \ldots, y_{n-1}), \ldots, C_{m-1}(y_o, \ldots, y_{n-1}))$ is in P then let $A'(x_o, \ldots, x_{n-1}) \to B'(C_o'(x_o, \ldots, x_{n-1}), \ldots, C'_{m-1}(x_o, \ldots, x_{n-1}))$ be in P';

(b) If $A(y_o, \ldots, y_{n-1}) \to fy_{i_o} \ldots y_{i_k}$ is in P, then let $A'(x_o, \ldots, x_{n-1}) \to x_{i_o} \ldots x_{i_k}$ be in P';

(c) If $A(y_o, \ldots, y_{n-1}) \to y_i$ is in P, then let $A'(x_o, \ldots, x_{n-1}) \to x_i$ be in P'.

It is obvious that for $t_1, t_2 \in W_\Sigma(X)_o$, $t_1 \underset{G}{\Longrightarrow} t_2$ via the rule $F(y_o, \ldots, y_{n-1}) \to t$ if and only if $\phi_\Omega(\mathrm{YIELD}_o(t_1)) \underset{G'}{\Longrightarrow} \phi_\Omega(\mathrm{YIELD}_o(t_2))$ via the corresponding rule in P'. So we have the result:

$L(G') = \phi_\Omega(\mathrm{YIELD}_o(L(G))$.

Corollary. Given a context free term grammar G over the alphabet $D_{()}(T(\Omega))$, we can effectively find an indexed grammar G' over Ω such that $\phi_\Omega(L(G')) = \mathrm{YIELD}_o(L(G))$.

Corollary. Let Ω be a many-sorted alphabet and let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a context free term grammar. Consider $G' = \langle \bigsqcup_{i \in I} \Omega_{\langle \lambda, i \rangle}, N', V, P', Z' \rangle$ where:

(i) $N'_n = \bigsqcup_{\ell(w)=n} \Omega_{\langle w,i \rangle}$ for each $n \in \underline{N}$;

(ii) $V = \{x_o, \ldots, x_{k-1}\}$ where k is the highest rank of any $A \in N$;

(iii) As in the theorem.

Then G' is a macro-grammar structure over $\bigsqcup_{i \in I} \Omega_{\langle \lambda, i \rangle}$ and L(G') is an indexed set.

Corollary. Let $G = \langle \Omega, \Sigma, P, Z \rangle$ be a context free term grammar. Then there is an effective procedure to decide whether $p \in P$ is useless or not.

Proof. The emptyness problem is solvable for the class of indexed sets (see Aho (1)). Use the previous corollary to get the desired result.

Example 2. Consider the indexed set of strings $L = \{a^n b^n c^n | n > 0\}$. $L = \mathrm{YIELD}_1(L(G))$ where G is the context free term grammar of Example IV.1.3b.

Theorem 7. Suppose Ω is a set and G is an indexed grammar over Ω. Then we can effectively find a regular grammar G' over the many-sorted alphabet $D_{()}(D_{()}(T(\Omega)))$ such that $\phi_\Omega(L(G)) = \mathrm{YIELD}_o(\mathrm{YIELD}_{\langle \lambda, o \rangle}(L(G')))$.

Conversely, given a regular grammar G over the alphabet $D_{()}(D_{()}(T(\Omega)))$, if L(G) is of sort $\langle \lambda, 0 \rangle$, then we can effectively find an indexed grammar G' over Ω such that $\phi_{\Omega}(L(G')) = YIELD_{o}(YIELD_{\langle \lambda, o \rangle}(L(G)))$.

<u>Proof</u>. An easy consequence of Theorems 5 and 6 and the Fundamental Theorem.

<u>Example 3</u>. Consider the regular grammar G" of Example V.1.1. L of Example 2 is then just $YIELD_{1}(YIELD_{1}(L(G")))$.

We remark that this is a very important result as it has 'reduced' a very complicated class of sets (at least in terms of its grammars) to a much simpler class of sets (albeit over a more complicated alphabet). Put another way, if we can refer to context free term sets as in some way characterising the derivations of indexed grammars, we have characterised the derivation trees of the derivation trees of indexed languages as recognizable sets.

Inspired by these ideas, in the general case of a many-sorted alphabet Σ, we may define the <u>indexed sets of terms</u> over Σ of type $\langle \lambda, \langle \lambda, i \rangle\rangle$ to be the images under the maps $YIELD_{\langle \lambda, \langle \lambda, i \rangle\rangle}$ and $YIELD_{\langle \lambda, i \rangle}$ (in that order) of some recognizable subset of $(W_{D_{()}(D_{()}(\Sigma))})_{\langle \lambda, \langle \lambda, i \rangle\rangle}$. More generally, we may define the indexed sets of terms over Σ of type $\langle \lambda, \langle w, i \rangle\rangle$ to be the image under the maps $YIELD_{\langle \lambda, \langle w, i \rangle\rangle}$ and $YIELD_{\langle w, i \rangle}$ (in that order) of some recognizable subset of $(W_{D_{()}(D_{()}(\Sigma))})_{\langle \lambda, \langle w, i \rangle\rangle}$. Even more generally, we may define the indexed sets of terms over Σ of type $\langle\langle w, i\rangle\langle w, v_{o}\rangle \ldots \langle w, v_{m-1}\rangle, \langle v, i\rangle\rangle$ to be the images under $YIELD_{\langle\langle w, i\rangle\langle w, v_{o}\rangle \ldots \langle w, v_{m-1}\rangle, \langle v, i\rangle\rangle}$ and $YIELD_{\langle v, i \rangle}$ (in that order) of some recognisable set in $(W_{D_{()}(D_{()}(\Sigma))})_{\langle\langle w, i\rangle\langle w, v_{o}\rangle \ldots \langle w, v_{m-1}\rangle, \langle v, i\rangle\rangle}$. Obviously an indexed set of type $\langle\langle w,i\rangle\langle w,v_0\rangle \ldots \langle w,v_{m-1}\rangle , \langle v,i\rangle\rangle$ is a set of derived operations. An indexed family of indexed sets of terms over Σ (note the two meanings of indexed) is indexed by $D_{()}(D_{()}(I))$. We can now state the important

Theorem 8 (Second Fundamental Theorem). Given an indexed family of indexed sets of terms L over the many-sorted alphabet Σ, we can effectively find an indexed family recognizable sets of terms L' over the alphabet $D_{()}(D_{()}(\Sigma))$ such that L = YIELD(YIELD(L')) and conversely. (Note that we have used the name 'YIELD' for both the homomorphisms $\text{YIELD}:W_{D_{()}(D_{()}(\Sigma))} \to D_{()}(W_{D_{()}(\Sigma)})$ and $\text{YIELD}:W_{D_{()}(\Sigma)} \to D_{()}(W_{\Sigma}))$.

We remark that the Second Fundamental Theorem does not prove that the class of context free sets of type $\langle w,i\rangle$ over Σ is a proper subclass of the class of indexed sets of type $\langle\lambda, \langle w,i\rangle\rangle$ over Σ. The following will rectify this point.

Theorem 9. The class of context free sets of type $\langle w,i\rangle$ over Σ is a proper subclass of the class of indexed sets of type $\langle\lambda, \langle w,i\rangle\rangle$ over Σ for any many-sorted alphabet Σ.

Proof. We know from the second Corollary of Theorem 6 that there is an effective procedure to obtain an indexed grammar over the set $\bigcup_{i\varepsilon I} \Sigma_{\langle\lambda,i\rangle}$ from the context free term grammar $G = \langle \Sigma, \Delta, P, Z\rangle$. We can extend this result in the following way: Let $fr:W_\Sigma \to (\bigcup_{i\varepsilon I} \Sigma_{\langle\lambda,i\rangle})^*$ be defined by:

(0) $fr(a) = a$ for $a \varepsilon \Sigma_{\langle\lambda,i\rangle}$, some $i \varepsilon I$;

and (1) $fr(ft_0 \ldots t_{n-1}) = fr(t_0) \ldots fr(t_{n-1})$ for $ft_0 \ldots t_{n-1} \varepsilon \hat{W}_\Sigma$.

(This is the definition of 'yield', 'leaf profile', etc., given in Mezei and Wright, Brainerd (1), Rounds (2), et al.) Then we claim $L(G') = fr(L(G))$. The proof is left to the interested reader. The proof of the theorem then proceeds by showing that there exists an indexed set L of type $\langle\lambda,\langle w,i\rangle\rangle$ over Σ such that fr(L) is not an indexed set of strings. We will show this only for the case $\Sigma = D_{()}(T(\Delta))$, Δ a set. The interested reader is referred to Turner for a proof of the general case.

Consider the alphabet $D_{()}(\Omega)$ of Example V.1.2 and the following context free grammar $G = \langle D_{()}(D_{()}(T(\Delta)), \Sigma', P, Z\rangle$ over the alphabet $D_{()}(D_{()}(T(\Delta))$ ($\Delta = \{a\}$ and Ω is $D_{()}(T(\Delta))$ less the 'redundant' individual symbols $\delta^1_{(0)}$ and $\delta^1_{(1)}$) where:

(i) $N_{\langle\lambda,\langle\lambda,0\rangle\rangle} = \{Z\}$, $N_{\langle\langle 1,1\rangle\langle 1,1\rangle,\langle 1,1\rangle\rangle} = \{E\}$;

(ii) $P = \{Z \to c_{\langle 10,10,0\rangle}{}^{*}\, c_{\langle 11,1,1\rangle}\, {+}aa\lambda,$

$Z \to c_{\langle 10,10,0\rangle}{}^{*}\, c_{\langle 1,1,1\rangle} E(c_{\langle 11,1,1\rangle} + \delta^1_{(\langle 1,1\rangle)}) \delta^1_{(\langle 1,1\rangle)})$
$a\lambda,$

$E(x) \to E(c_{\langle 1,1,1\rangle}\, xx),$

$E(x) \to c_{\langle 1,1,1\rangle}\, xx\}$ for $x \in X_{\langle 1,1\rangle}\}$.

Then $L' = YIELD_{\langle\lambda,0\rangle}(L(G))$ is an indexed set of terms of type $\langle\lambda,\langle\lambda,0\rangle\rangle$ over $D_{()}(T(\Delta)) = \Omega$. Then by the above discussion, fr(L') is an indexed set of strings. But it is clear that

$L'' = fr(L') = \{a^{2^{2^{n}}}\lambda \mid n \geq 0\}$ (λ here is not the empty string). L'' is not an indexed set of strings, so L' is not a context free set of terms (of type $\langle\lambda, 0\rangle$ over Ω).

Note the fact that L', as a set, is also a subset of $D_{()}(W_{D_{()}(T(\Delta))})_{\langle w,i\rangle}$ (and so of $W_{D_{()}(T(\Delta))}(X_w)_i$) for any $\langle w,i\rangle \in D_{()}(\{0,1\})$. This is clear since $\lambda \notin L'$ and $t \in L'$ is just a derived operation of type $\langle w,i \rangle$ in which none of X_w appears. There exists a context free set of terms $\overline{L}$ over $D_{()}(D_{()}(T(\Delta)))$ such that $YIELD_{\langle w,i \rangle}(\overline{L}) = L'$. We again see that fr(L') is not an indexed set of strings (as above) and so we have proved our result for any $\langle w,i \rangle$.

VI.3. Homomorphism with Respect to Composition

Let Ω and Δ be sets. A homomorphism with respect to concatenation (of the conventional theory) $\phi' : \Omega^* \to \Delta^*$ is defined as the function generated by the assignments:

(i) $\phi_o(\lambda) = \lambda$ and

(ii) $\phi_1(a) = w \in \Delta^*$ for $a \in \Omega$.

(The algebras under consideration are, of course, the algebras with the single binary operation of concatenation and carriers Ω^* and Δ^*.)
Under the equivalence in the Remark at the end of Chapter II, ϕ' is seen to be the component $\overline{\phi}_o$ of the map $\overline{\phi} : D_{V_{T(\Omega),T(\Delta)}}(W_{T(\Omega)}) \to D_{V_{T(\Omega),T(\Delta)}}(W_{T(\Delta)})$ generated by the assignments (i) and (ii) above (again leaving out the 'redundant' projection operations).

It is a classical result of the conventional theory that if Ω and Δ are as above, $\phi' : \Omega^* \to \Delta^*$ is a homomorphism with respect to concatenation and $U \subseteq \Omega^*$, $V \subseteq \Delta^*$ are context free (recognizable) sets, then $p\phi'(U)$ and $(p\phi')^{-1}(V)$ are context free (recognizable) as well.
Motivated by this we state

Theorem 1. Let Ω and Δ be many-sorted alphabets, $\overline{\phi} ; D_{V_{\Omega,\Delta}}(W_\Omega) \to D_{V_{\Omega,\Delta}}(W_\Delta)$ be a homomorphism with respect to composition, and $U \subseteq W_\Omega$ be a family of context free sets. Then so is $p\overline{\phi}(U)$.

Proof. Suppose $U = L(G)$ where $G = \langle \Omega, \Sigma, P, Z \rangle$ is a normal form grammar. Then let $G' = \langle \Delta, \Sigma', P', Z \rangle$ be the grammar obtained by forming the alphabet Σ' in the obvious way and replacing the production $F(x_o, \ldots, x_{n-1}) \to \nu_f$ in P' for $F(x_o, \ldots, x_{n-1}) \to fx_{i_o} \ldots x_{i_k}$ in P where $\nu_f = \phi_{<w,i>}(f)$ for $f \in \Omega_{<w,i>}$. All other productions of P' are the same as those of P. G' is a context free grammar (in standard form). Note that there is a one-one correspondence between productions in P and those in P'.

We can extend $\overline{\phi}$ to $\overline{\phi}: D_{V_{\Sigma,\Sigma'}}(W_\Sigma) \to D_{V_{\Sigma,\Sigma'}}(W_{\Sigma'})$ by defining $\phi_{<w,i>}(F) = F$ for $F \in N_{<w,i>}$. Remark that $t \underset{G}{\Longrightarrow} t'$ via a production in P if and only if $\overline{\phi}(t) \underset{G'}{\Longrightarrow} \overline{\phi}(t')$ via the corresponding production in P'. It can now easily be shown that $p\overline{\phi}(U) = L(G')$.

Corollary. If in the statement of the theorem, U is recognizable, then so is $p\overline{\phi}(U)$.

Proof. The proof is similar to the above except for the fact that we first obtain a regular system G' from the regular grammar G for U. We then invoke the results of the remark at the end of Chapter III, Section 2.

Unfortunately, the closure of the class of families of context free sets under the operation of inverse homomorphism is not immediately obvious. It is hoped that this open problem will soon be solved. If it is, then the class of (indexed families of) context free sets of terms would have properties analogous to a Full Abstract Family of (String) Languages (see Ginsburg (2)). (A full AFL is a class of sets (of strings) closed under the operations of union, complex product, Kleene closure, intersection with recognizable sets, homomorphism and inverse homomorphism.)

VI.4. An Alternative Definition

One of the characteristics of recognizable sets is that they can be defined independently of any generating (or, equivalently, any fixed point) system. It was shown in Chapter III, Section 2 that the class of recognizable sets of terms over Ω is equivalent to the class of unions of the classes of finite congruences over W_{Ω}. Motivated by the Fundamental Theorem we make the following definition: A context free set of terms over a many-sorted alphabet Ω is the homomorphic image (under YIELD) of the union of classes of some finite congruence q over $W_{D_{()}(\Omega)}$. Motivated by the Second Fundamental Theorem, we define indexed sets of terms over Ω to be the homomorphic images (under two separate YIELD functions) of the union of classes of some finite congruence q' over $W_{D_{()}(D_{()}(\Omega))}$.

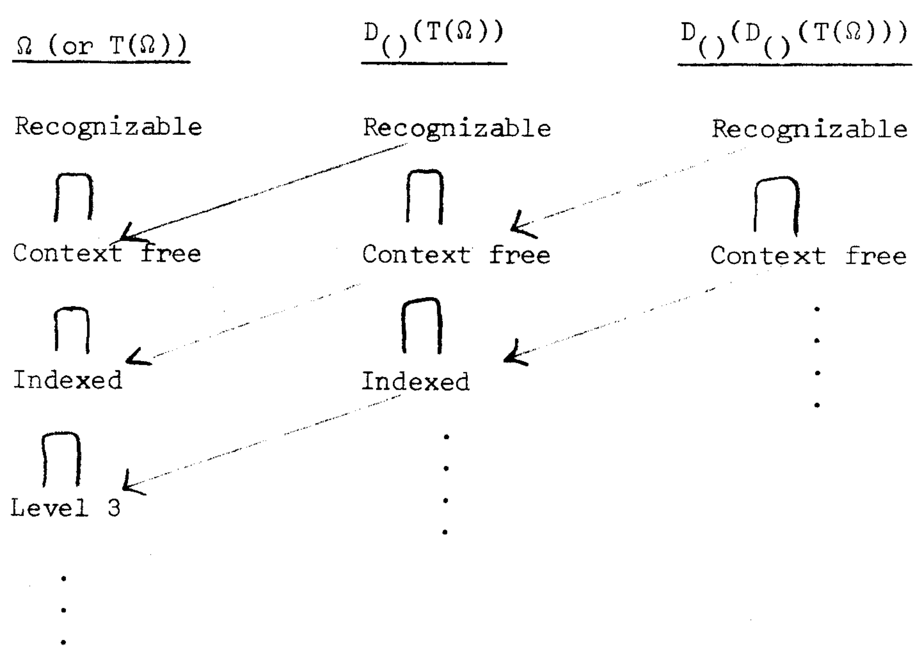
VII CONCLUSION

We have attempted in this study to achieve two main objectives:

(i) To place the study of formal languages in a setting general enough to facilitate the consolidation of existing results and to open new avenues of research;

(ii) To prove within this framework what we have called the Fundamental Theorem of Formal Languages: specifying the fundamental nature of recognizable sets.

In a sense, the Fundamental Theorem shows that we only need to 'know' about recognizable sets. Other classes of sets are just homomorphic images of certain classes of recognizable sets. Now, consider the grammar G of Theorem VI.2.9. We know YIELD(YIELD(L(G))) = $\{a^{2^{2^n}} \mid n > 0\}$ which is not an indexed set of strings. Let us depict in pictorial form what this and the Fundamental Theorem suggest for a set (string alphabet) Ω:



The inclusions indicated are of classes and the arrow indicates equivalence of classes via the YIELD operation (homomorphism). We have called the image under YIELD of an indexed set of terms over $D_{()}(T(\Omega))$ a Level (of complexity) 3 set (as opposed to 'type' used in the Chomsky classification of language classes). According to this method of nomenclature, recognizable sets over Ω are level of complexity 0, context free sets over Ω are level of complexity 1, etc. We might call any set over Ω of level of complexity n (any $n \in \underline{N}$), a set over Ω of level of generalisation 0. Any set over $D_{()}(T(\Omega))$ of level of complexity n (any $n \in \underline{N}$) is a set over Ω of level of generalisation 1. And so on.

Our diagram suggests that it may be extended infinitely in two dimensions. Thus any class within this extension can be pinpointed by two coordinates: its level of generalisation and its level of complexity within that. Thus each increase in the level of generalisation 'reduces' the level of complexity by one! Note that we need not have started with a string alphabet. Any many-sorted alphabet would have done just as well.

We note that, at all levels of generalisation greater than zero, all non-individual symbols in our many-sorted alphabets were assigned typed (in the logical sense) composition operations. Level of generalisation increased with the 'level of the logical types' of the operator symbols. We speculate that a third 'dimension' may be added to the diagram by considering many-sorted alphabets which will have logically typed application operations assigned to its symbols. (Note that composition is a particularly simple form of application.) Note the connection with Scott's models of the lambda calculus.

As noted, the comments above are just speculation; on the other hand, we hope that they are not idle speculation. They demand a lot of study before confirmation can be obtained. Here are some other questions that demand answers: Are there classes of automata of different levels of complexity and generalisation to match our classes of sets? What light, if any, does this approach to formal languages shed on the study of natural languages and programming languages? More specifically, can we prove the closure of a class of sets under the operation of inverse homomorphism?

We believe both our stated aims to be achieved.

REFERENCES

AHO A.V., (1): 'Nested Stack Automata', Journal of the ACM, Vol.16, No.3.

AHO A.V., (2): 'Indexed Grammars - An Extension of Context Free Grammars', Journal of the ACM, Vol.15, No.4.

ARBIB M.A. and GIVE'ON Y., 'Algebra Automata I: Parallel Programming as a Prolegomena to the Categorical Approach', Information and Control, Vol.12, pp.331-345.

BENABOU J., 'Structures Algebriques dans les Categories', Doctoral Dissertation, University of Paris, 1966.

BIRKHOFF G. and LIPSON J.B., 'Heterogeneous Algebras', Journal of Combinatorial Theory, Vol.8, pp.115-133.

BRAINERD W.E., (1): 'Tree Generating Regular Systems', Information and Control, Vol.14, pp.217-231.

(2): 'The Minimalization of Tree Automata', Information and Control, Vol.13, pp.484-491.

BUCHI J.R., (1): 'Weak Second-order Arithmetic and Finite Automata', Z.Math.Logik Grundlagen Math., Vol.6, pp.66-92.

(2): 'On a Decision Method in Restricted Second-order Arithmetic', Proc.1960 International Congress on Logic, Methodology, and the Philosophy of Science, pp.1-11.

(3): 'Regular Canonical Systems', Archiv fur Mathematische Logik und Grundlagen-forschung', Vol.6, pp.91-111.

BUCHI J.R. and ELGOT C.C., 'Decision Problems of Weak Second-order Arithmetic and Finite Automata', Notices of the American Math. Soc., Vol.5.

BURSTALL R.M. and LANDIN P.J., 'Programs and their Proofs: An Algebraic Approach', Machine Intelligence 4, Edinburgh University Press, 1969.

COHN P.M., (1): 'Universal Algebra', Harper and Row, 1965.

(2): 'The Embedding of Firs in Skew Fields', Proc.London Mathematical Society', Vol.23, pp.193-213.

CHOMSKY N., (1): 'On Certain Formal Properties of Grammars', Information and Control, Vol.2, pp.137-167.

(2): 'Context Free Grammars and Pushdown Storage', Quart.Prog.Dept.No.65, M.I.T.Res.Lab.Elect., pp.187-194.

CHOMSKY N. and MILLER G.A., 'Finite State Languages', Information and Control, Vol.1, pp.91-112.

CHURCH A., 'The Calculi of Lambda-Conversion', Princeton University Press, 1941.

CURRY H.B. and FEYS R., 'Combinatory Logic', North-Holland, 1958.

DAVIS M., 'Computability and Unsolvability', McGraw-Hill, 1958.

DODGSON C.L., 'The Walrus and the Carpenter', in The Complete Works, Nonesuch Press, 1939.

DONER J.E., 'Tree Acceptors and Some of Their Applications', System Development Corp., Scientific Report No.8, 1967.

EILENBERG S. and WRIGHT J.B., 'Automata in General Algebras', Information and Control, Vol.11, pp.452-470.

FISCHER M.J., (1): 'Grammars with Macro-like Productions', Doctoral Dissertation, Harvard University, 1968.

(2): 'Grammars with Macro-like Productions', Proc.9th I.E.E.E.Symp. on Switching and Automata Theory, 1968.

GINSBURG S., 'The Mathematical Theory of Context-Free Languages', McGraw-Hill, 1966.

GINSBURG S. and GREIBACH S.A., 'Abstract Families of Languages', Proc. 8th I.E.E.E.Symp.on Switching and Automata Theory, 1967.

GRATZER G., (1): 'Universal Algebra', D.van Nostrand, 1968.

(2): 'Free Σ-structures', Transactions of the American Mathematical Society, Vol.135, pp.517-542.

GROSS M. and LENTIN A., 'Introduction to Formal Grammars', George Allen and Unwin Ltd., 1970.

HALL P., 'Some Word Problems', Journal of the London Mathematical Society, Vol.33, pp.482-496.

HALMOS P.R., 'Naive Set Theory', D.Van Nostrand, 1961.

HIGGINS P.J., 'Algebras with a Scheme of Operators', Math.Nachr. Vol.27.

HOPCROFT J.E. and ULLMAN J.D., 'Formal Languages and their Relation to Automata', Addison-Wesley, 1969.

KAPLAN D.M., 'The Formal Theoretic Analysis of Strong Equivalence for Elemental Programs', Doctoral Dissertation, Stanford University, 1968.

KLEENE S.C., 'Representation of Events in Nerve Nets and Finite Automata', Automata Studies, Princeton University Press, 1956.

LANDIN P.J., (1): 'The Mechanical Evaluation of Expressions', Computer Journal, Vol.6, pp.308-320.

(2): 'A Correspondence between ALGOL 60 and Church's Lambda Notation', Communications of the ACM, Vol.8, pp.89-101,; 158-165.

(3): 'A λ-calculus Approach', Advances in Programming and non-numerical Computation, Pergamon Press, 1966.

(4): 'A Program Machine Symmetric Automata Theory', Machine Intelligence 5, Edinburgh University Press, 1970.

LAWVERE F.W., (1): 'Functorial Semantics of Algebraic Theories', Proc. National Academy of Sciences, Vol.50, pp.869-872.

(2): 'Algebraic Theories, Algebraic Categories and Algebraic Functors', The Theory of Models, Proc.1963 Int.Symp. at Berkeley, North-Holland, 1965.

MAIBAUM T.S.E., (1): 'The Characterisation of the Derivation Trees of Context Free Sets of Terms as Regular Sets', Proc.13th I.E.E.E. Symp. on Switching and Automata Theory, 1972.

(2): 'A Generalized Approach to Formal Languages', to be published in the Journal of Computer and System Sciences.

MAL'CEV A.I., 'Algorithms and Recursive Functions', Wolters-Noordhoff Publishing, 1970.

MEZEI J. and WRIGHT J.B., 'Generalised ALGOL-like Languages', I.B.M. Research Paper, RE1528, 1965.

PIAGET J., 'Structuralism', Routledge and Kegan Paul, 1971.

RABIN M.O. and SCOTT D., 'Finite Automata and their Decision Problems', I.B.M.Journal of Research and Development, Vol.3, pp.115-125.

ROGERS H. Jr., 'Theory of Recursive Functions and Effective Computability', McGraw-Hill, 1967.

ROUNDS W.C., (1): 'Mappings and Grammars on Trees', Mathematical Systems Theory, Vol.4, No.3.

(2): 'Tree Oriented Proofs of Some Theorems on Context Free and Indexed Languages', Proc.ACM Symp. on Theory of Computing, 1970.

(3): 'Trees, Transducers and Transformations', Doctoral Dissertation, Stanford University, 1968.

SCOTT D., (1): 'The Lattice of Flow Diagrams', Tech.Monograph PRG-3, Programming Research Group, Oxford University Computing Laboratory, 1970.

(2): 'Continuous Lattices', Tech.Monograph PRG-7, Programming Research Group, Oxford University Computing Laboratory, 1971.

THATCHER J.W., (1): 'Generalised^2 Sequential Machines', Journal of Computer and System Sciences, Vol.4, pp.339-367.

(2): Private Communication, 1972.

(3): 'Characterising Derivation Trees of Context Free Grammars Through a Generalisation of Finite Automata Theory', Journal of Computer and System Sciences, Vol.1, pp.317-322.

THATCHER J.W. and WRIGHT J.B., 'Generalised Finite Automata Theory with an Application to a Decision Problem of Second Order Logic', Mathematical Systems Theory, Vol.2, No.1.

THUE A., 'Die Losung eines Spezialfalles eines Generellen Logischen Problems', Videnskabs-Selskabets Skrifter, I.Math.-Naturv. Klasse, No.8, 1910.

TURNER R., Doctoral Dissertation, University of London, 1973.