The Basic Algebraic Structures in Categories of Derivations*

DAVID B. BENSON

Computer Science Department, Washington State University,
Pullman, Washington 99163

The basic algebraic structures within the categories of derivations determined by rewriting systems are presented. The similarity congruence relation in categories of derivations is given in three versions. The syntax category is formed by taking derivations modulo similarity. This category is a free strict monoidal category, a simple form of a 2-category. The syntax category is central to the study of rewriting systems, morphisms in the category generalizing the notion of "derivation tree," so a detailed development is given. Griffith's interchange operators on derivations form a 2-category over a category of derivations. Representability of a similarity class is defined and shown to imply the existence of group of operators on the class, induced by interchanges. Uniform representability of rewriting systems is defined and shown to imply that the set of left divisors of each derivation in the syntax category is a distributive lattice.

1. INTRODUCTION

This report presents the basic algebraic facts about categories of derivations and related systems. The purpose is to definitively establish the algebraic framework for further studies of the syntax and parsing of languages and of the general theory of translation, compilation, and interpretation. The results proved are elementary in the sense that only basic knowledge about categories, partial functions, groups, partial orders and lattices is used. The required knowledge of categories is given here, but readers unfamiliar with categorical algebra may wish to consult MacLane (1971). The next few paragraphs summarize the contents.

In Section 2 the notation and the elements of the version of categorical algebra employed are set out. The notions of indexed rewriting systems, the

* Research supported in part by Grant GJ-1171 from the National Science Foundation.
derivations obtained from such systems, and their categoricity are presented in Section 3. The congruence relation of similarity between derivations is given in two forms in Section 4. In Section 5 the quotient category of derivations modulo similarity is shown to be an $x$-category (strict monoidal category). This category is sufficiently central to the study of the syntax as well as the semantics of formal languages, programming languages, etc., to warrant designation as the syntax category. [We abandon the terminology of (Benson, 1970)]. The construction used here is different that in Hotz (1966, 1965) and all details are given except for the proof that the syntax category is a free $x$-category.

Operators on derivations which interchange the position of rule applications are shown to form a category in Section 6. Composition of derivations induce a second composition on the interchanges, so that the interchange category is an instance of a 2-category. The details are similar to those in Section 5 and are not given. In Section 7 it is shown that the interchange 2-category presents categorically the data of the similarity congruence.

In Section 8 representable similarity classes are defined and characterized. The interchange operators are shown to induce a group on each representable similarity class.

Section 9 is devoted to providing, for the first time, the exact condition on a rewriting system so that each derivation in the syntax category forms a distributive lattice under divisibility. This condition, called uniform representability, is a slight extension of Chomsky's type 0 grammars so it includes essentially every rewriting system of practical interest. The proof is rather delicate, requiring almost everything established in the first eight sections. Further, the proof contains some information of independent interest. These results seem likely to suggest further remarkable facts about formal language parsing, translation, and interpretation.

2. Notation and Definitions

The basic elements of the relevant theory are reviewed to establish notation. For ease of reference, all definitions, lemmas, propositions, and theorems set forth are numbered in common, without regard to sections.

The application of a function, $F: A \rightarrow B$, to an element of its domain, $a$, will be variously denoted by $F(a)$, $aF$, or by the barred arrow notation, $F: a \rightarrow b$. The choice of notation will remain consistent for each particular class of functions considered.

The image of a function $F: A \rightarrow B$ is the set $\text{Im}(F) = \{b \mid aF = b\}$. The
restriction of a function to a subset \( A' \) of its domain is denoted \( F \mid A' \).
Similarly, if \( R \subseteq A \times B \) is a relation, the restriction of \( R \) to the subset \( Q \subseteq A \times B \) is denoted \( R \mid Q \) and is equal to \( R \cap Q \).

\( N = \{1, 2, \ldots\} \) is the set of natural numbers and \( n = \{1, 2, \ldots, n\} \) is an initial segment of \( N \).

If \( F: A \rightarrow B \) is a partial function, then we say that \( F \) applies to \( a \) or that \( aF \) is defined whenever \( a \) is in the proper domain of \( F \).

Composition of functions and partial functions will always be written in linguistic order, so that for \( F: A \rightarrow B, G: B \rightarrow C, F \circ G, \) and \( FG \) mean first \( F \), then \( G \). Composition of morphisms in categories will also be written in linguistic order.

The definition of category we use is quite restrictive, since we insist that the objects and morphisms are sets, not classes. However, such small categories are just what we need, and the definition is convenient to work with.

**Definition 1.** A category is a 5-list, \( C = (O, M, d, c, o) \) such that

(i) \( O \) is a set of objects,
(ii) \( M \) is a set of morphisms,
(iii) \( d: M \rightarrow O \) is a function, the domain function,
(iv) \( c: M \rightarrow O \) is a function, the codomain function,
(v) \( o: M^2 \rightarrow M \) is a partial binary function, the composition operator.

These entities are subject to the following axioms.

(1) For all \( x_1, x_2 \in M \), \( x_1 \circ x_2 \) is defined iff \( c(x_1) = d(x_2) \).
(2) For all \( x_1, x_2 \in M \), if \( x_1 \circ x_2 \) is defined, then \( d(x_1 \circ x_2) = d(x_1) \) and \( c(x_1 \circ x_2) = c(x_2) \).
(3) For all \( x_1, x_2, x_3 \in M \) if \( (x_1 \circ x_2) \circ x_3 \) is defined then \( (x_1 \circ x_2) \circ x_3 = x_1 \circ (x_2 \circ x_3) \).
(4) For each \( A \in O \) there exists \( 1_A \in M \) such that
   (4.1) If \( x \circ 1_A \) is defined then \( x \circ 1_A = x \),
   (4.2) If \( 1_A \circ x \) is defined then \( 1_A \circ x = x \),
   (4.3) \( d(1_A) = c(1_A) = A \).

We read the axioms as follows: (1) The composition of the morphisms is defined iff the codomain of the first is the domain of the second. (2) The domain of a composite is the domain of its initial morphism and similarly for codomains. (3) Composition is associative wherever defined. (4) There is an identity morphism for each object.
PROPOSITION 2. There is a unique identity for each object. □

The box, □, denotes the end of as much proof as will be given.

The set of all the morphisms in category C with domain A and codomain B is denoted \( \text{Mor}_C(A, B) \).

**Definition 3.** A morphism \( x \) is epic if for all \( y, z \) such that \( x \circ y = x \circ z \), \( y = z \). A morphism \( x \) is monic if for all \( v, w \) such that \( v \circ x = w \circ x \), \( v = w \).

**Definition 4.** Let \( C = (O, M, d, c, \circ) \) be a category. If

\[
C' = (O', M', d', c', \circ')
\]

is a category such that \( O' \subseteq O, M' \subseteq M \), and \( d', c', \circ' \) are restrictions of \( d, c, \circ \), then \( C' \) is a subcategory of \( C \). \( C' \) is a full subcategory of \( C \) if for each \( A, B \in O' \), \( \text{Mor}_C(A, B) = \text{Mor}_{C'}(A, B) \).

The same notation, \( d, c, \circ \), for the domain, codomain, and composition functions is used in common for several categories, the particular functions denoted being determined by context.

**Definition 5.** (Benabou, 1965, 1967; Palmquist, 1971). A 2-category is a 9-list \( C = (O, O', M, d, c, \circ, d', c', \circ') \) such that

(i) \( (O, M, d, c, \circ) \) is a category with set of identities \( \text{Id} \),

(ii) \( (O', M, d', c', \circ') \) is a category with set of identities \( \text{Id}' \),

(iii) \( (O \cap O', \text{Id}', d, c, \circ) \) is a subcategory of \( (O, M, d, c, \circ) \),

(iv) \( (O \cap O', \text{Id}, d', c', \circ') \) is a subcategory of \( (O', M, d', c', \circ') \),

(v) For all \( x_1, x_2, y_1, y_2 \in M \), \( (x_1 \circ x_2) \circ' (y_1 \circ y_2) = (x_1 \circ' y_1) \circ (x_2 \circ' y_2) \) whenever both sides are defined.

(vi) \( O' \subseteq O, \text{Id}' \subseteq \text{Id} \), that is, every \( \circ' \)-identity is a \( \circ \)-identity.

**Definition 6.** (Hotz, 1965, 1966). An \( \lambda \)-category is a 2-category \( C = (O, O', M, d, c, \circ, d', c', \circ') \) such that \( O' \) and \( \text{Id}' \) are singleton sets. (So that \( \circ' \) is everywhere defined.)

This construct is also called a strict monoidal category (MacLane, 1971), defined by requiring that the composition \( \circ' \) be a strictly associative bifunctor on the category \( (O, M, d, c, \circ) \). The composition \( \circ' \) is often written \( \times, \otimes \), or even \( \Box \). In syntax categories \(+\) is used to denote this composition, basically concatenation.
DEFINITION 7. Let \( C_1 = (O_1, M_1, d, c, o) \) and \( C_2 = (O_2, M_2, d, c, o) \) be categories. A functor \( F: C_1 \to C_2 \), from \( C_1 \) to \( C_2 \), is a pair of functions,

(i) \( F_O: O_1 \to O_2 \), the object function, and
(ii) \( F_M: M_1 \to M_2 \), the morphism function, satisfying the requirements

1. \( F(1_A) = 1_{F(A)} \) for all identities \( 1_A \in M_1 \).
2. \( F(x_1 \circ x_2) = F(x_1) \circ F(x_2) \) for all \( x_1, x_2 \in M_1 \) such that \( x_1 \circ x_2 \) is defined.

The subscripts on the functor are not written since it is always possible to tell from context whether the object function or the morphism function is intended. Any functor, \( F: C \to C \), taking a category to itself is called an endofunctor.

PROPOSITION 8. Let \( F: C_1 \to C_2 \) be a functor. For all morphisms \( x \) of \( C_1 \),

\[
F(d(x)) = d(F(x)) \quad \text{and} \quad F(c(x)) = c(F(x)).
\]

Proof. \( 1_{d(F(x))} \circ F(x) = F(x) = F(1_{d(x)} \circ x) = F(1_{d(x)}) \circ F(x) = 1_{F(d(x))} \circ F(x). \) This calculation demonstrates that \( c(1_{d(F(x))}) = c(1_{F(d(x))}). \) Since identities are unique, \( 1_{d(F(x))} = 1_{F(d(x))} \) and then \( d(F(x)) = F(d(x)). \) Dually for codomains. \( \square \)

DEFINITION 9. Let \( C_i = (O_i, M_i, d, c, o), \) \( i = 1, 2, 3, \) be three categories. The product category \( C_1 \times C_2 \) has objects \( O_1 \times O_2 \), morphisms given by \( \text{Mor}_{C_1 \times C_2}((A, A'), (B, B')) = \text{Mor}_{C_1}(A, B) \times \text{Mor}_{C_2}(A', B') \), the domain, codomain, and composition functions of \( C_1 \times C_2 \) are determined by the component functions in \( C_1 \) and \( C_2 \). A functor \( F: C_1 \times C_2 \to C_3 \) is called a bifunctor from \( C_1 \) and \( C_2 \) to \( C_3 \).

DEFINITION 10. Let \( F_1, F_2 \) be functors from \( C_1 \) to \( C_2 \). A function from the objects of \( C_1 \) to the morphisms of \( C_2 \), \( N: O_1 \to M_2 \), is a natural transformation from \( F_1 \) to \( F_2 \) if for all \( x: A \to B \) morphisms of \( C_1 \),

\[
F_1(x) \circ N(B) = N(A) \circ F_2(x).
\]

DEFINITION 11. Let \( C = (O, M, d, c, o) \) be a category. For each \( A, B \in O \) let \( \equiv_{A,B} \subseteq (\text{Mor}_C(A, B))^2 \) be an equivalence relation. Let \( \equiv = \bigcup_{A,B} \equiv_{A,B} \). If it is the case that \( x_1 \equiv x_2, y_1 \equiv y_2 \) imply \( x_1 \circ y_1 \equiv x_2 \circ y_2 \) for all composable \( x_1, y_1 \), then \( \equiv \) is a congruence relation for the category \( C \). The corresponding quotient category has as its morphisms the congruence classes under \( \equiv \). The set of all congruence classes is denoted \( M/\equiv \). The domain, codomain, and composition functions for the quotient category are determined by representatives.
The congruence \( \equiv \) establishes the natural epifunctor from \( C = (0, M, d, c, \circ) \) to \( C/\equiv = (0, M/\equiv, d, c, \circ) \), the identity on objects and taking each morphism \( x \) to its congruential equivalence class \([x]\).

Lower-case greek letters will denote just strings over the alphabet \( \Sigma \). \( \Sigma \) generates the free monoid \((\Sigma^*, +, \lambda)\). The \(+\) may be elided so that \( \alpha \beta = \alpha + \beta \). \( l(\alpha) \) denotes the length of \( \alpha \). \( \Sigma^+ = \Sigma^* - \{\lambda\} \) is the set of all nonnull strings. Elements of \( \Sigma^* \times \Sigma^* \) are written \( \alpha \rightarrow \beta \) and are called productions or rewrite rules.

3. The Category of Derivations

The generalization of rewriting systems to indexed rewriting systems is rather more convenient for our development.

**Definition 12.** The pair \((\Sigma, P)\) is an indexed rewriting system if \( \Sigma \) is a set and \( P: J \rightarrow \Sigma^* \times \Sigma^* \) is a function from an indexing set \( J \) to rewrite rules. Write \( P(j) = \alpha \rightarrow \beta \) for the values of the indexing function \( P \).

The notation \((\Sigma, P)\) will always refer to an indexed rewriting system.

**Definition 13.** A derivation of length \( n \) in \((\Sigma, P)\) with domain \( \theta_1 \) and codomain \( \theta_{n+1} \) is a triple of finite sequences, \( x = (\text{pr}^x, \text{r}^x, \text{k}^x) \), such that

(i) \( \text{pr}^x = (\theta_1, \ldots, \theta_{n+1}) \) is a sequence of length \( n + 1 \) of strings in \( \Sigma^* \), called the proof sequence.

(ii) \( \text{r}^x = (r_1, \ldots, r_n) \) is a sequence of length \( n \) of indices in \( J \), called the rule sequence.

(iii) \( \text{k}^x = (\mu_1, v_1, \ldots, \mu_n, v_n) \), is a sequence of length \( n \) of pairs of strings in \( \Sigma^* \), called the context sequence. The underscore notation, \( \mu, v \) rather than \( (\mu, v) \), is used to emphasize that rewriting of strings occurs between the contexts in the position of the underscore.

(iv) For each \( i, 1 \leq i \leq n, \theta_i = \mu_i \circ v_i \), \( \theta_{i+1} = \mu_i \beta v_i \), where \( P(r_i) = \alpha \rightarrow \beta \).

Let \( D \) denote the set of all derivations in \((\Sigma, P)\), including those of zero length. For \( x \in D \) write \( d(x) \) for the domain and \( c(x) \) for the codomain, \( d: D \rightarrow \Sigma^* \), \( c: D \rightarrow \Sigma^* \). If \( d(x) = \theta \), \( c(x) = \psi \) write \( x: \theta \rightarrow \psi \). \( l(x) \) denotes the length of \( x \). For \( \theta \in \Sigma^* \) denote by \( 1_{\theta} \) the length zero derivation \( 1_{\theta} = ((\theta), ()), () \).

**Definition 14.** Let \( x, y \in D \) be

\[
x = ((\theta_1, \ldots, \theta_{n+1}), (r_1, \ldots, r_n), (\mu_1, v_1, \ldots, \mu_n, v_n))
\]
and \( y = ((\psi_1, \ldots, \psi_{m+1}), (s_1, \ldots, s_m), (\sigma_1, \ldots, \sigma_{m+1})) \). The composition of \( x \) and \( y \) is defined iff \( c(x) = d(y) \) and if defined is the derivation \( x \circ y = ((\theta_1, \ldots, \theta_n, \psi_1, \ldots, \psi_{m+1}), (\tau_1, \ldots, \tau_n, s_1, \ldots, s_m), (\mu_1, \ldots, \mu_n, \sigma_1, \ldots, \sigma_{m+1})) \).

With this data, \( D = (\Sigma^*, D, d, c, \circ) \) is a category, called the category of derivations in \( (\Sigma, P) \).

4. Similar Derivations

Two derivations in a category of derivations \( D \) are similar, or inessentially different, in the following intuitive way. If \( x \in D \) is such that the rule application \( r_i \) and \( r_{i+1} \) do not overlap in the sense that \( r_i \) and \( r_{i+1} \) apply to distinctly separate substrings, then application of these two rules in the opposite order results in another derivation \( y \), similar to \( x \). Although the definition of derivation requires that the application of rules be strictly sequential, the independent rewriting of \( r_i \) and \( r_{i+1} \) may be considered to occur in parallel,

\[
\begin{align*}
\mu & \quad + \quad \varepsilon & \quad + \quad \phi & \quad + \quad \gamma & \quad + \quad \nu \\
\mu & \quad + \quad \beta & \quad + \quad \phi & \quad + \quad \delta & \quad + \quad \nu
\end{align*}
\]

for \( P(r_i) = \alpha \rightarrow \beta \) and \( P(r_{i+1}) = \gamma \rightarrow \delta \). The two sequentializations of this parallel rewriting give the distinct derivations \( x \) and \( y \). The closure of this basis to an equivalence relation determines similarity.

The definition of similar derivations can be given in several equivalent ways. One is by interchanges, considered in subsequent sections. A second is by semantic considerations (Benson, 1970). A third is by congruences on the category \( D \). Using the third approach, two alternative definitions using congruences are shown equivalent.

The congruential definition of similarity requires, as preliminaries, the consideration of a collection of endofunctors on the category of derivations \( D = (\Sigma^*, D, d, c, \circ) \).

**Definition 15.** For each \( \sigma \in \Sigma^* \), define the pair of functions \( \sigma \rightarrow : \Sigma^* \rightarrow \Sigma^* \) and \( \sigma \rightarrow : D \rightarrow D \) as follows:

(i) \( \sigma \rightarrow (\omega) = \sigma + \omega = \sigma \omega, \) for \( \omega \in \Sigma^* \).

(ii) For \( x \in D \), \( x = ((\theta_1, \ldots, \theta_{n+1}), (r_1, \ldots, r_n), (\mu_1, \ldots, \mu_n, \nu_n)) \),

\( \sigma \rightarrow (x) = \sigma + x \) is the derivation

\( \sigma + x = ((\sigma \theta_1, \ldots, \sigma \theta_{n+1}), (r_1, \ldots, r_n), (\sigma \mu_1, \ldots, \sigma \mu_n)). \)
Dually, for each $\rho \in \Sigma^*$, define the pair of functions $-+\rho: \Sigma^* \to \Sigma^*$ and $-+\rho: D \to D$ as follows:

(i) $-+\rho(\omega) = \omega + \rho = \omega\rho$, for $\omega \in \Sigma^*$.

(ii) For $x \in D$, $x = ((\theta_1, \ldots, \theta_{n+1}), (\tau_1, \ldots, \tau_n), (\mu_1 \nu_1, \ldots, \mu_n \nu_n))$ $-+\rho(x) = x + \rho$ is the derivation

$$x + \rho = ((\theta_1\rho, \ldots, \theta_{n+1}\rho), (\tau_1, \ldots, \tau_n), (\mu_1\nu_1\rho, \ldots, \mu_n\nu_n)).$$

**Proposition 16.** The function pairs $\sigma -+$ and the function pairs $-+\rho$ are endofunctors, $\sigma -+: D \to D$ and $-+\rho: D \to D$.

**Proof.** $\sigma + 1_\theta = 1_{o+\theta} = 1_{o\theta} \cdot \sigma + (x_1 \circ x_2) = (\sigma + x_1) \circ (\sigma + x_2)$. Dually for $-+\rho$. $\Box$

For subsequent use, we note that the endofunctors on opposite sides commute while endofunctors on the same side associate. For each $\sigma, \rho \in \Sigma^*$, $(\sigma + -) \circ (-+\rho) = (-+\rho) \circ (\sigma + -)$. That is, for each $\omega \in \Sigma^*$, $(\sigma + \omega) + \rho = \sigma + (\omega + \rho)$, and for each $x \in D$, $(\sigma + x) + \rho = \sigma + (x + \rho)$. We write the composite endofunctor as $\sigma -+\rho$. For each $\sigma, \rho \in \Sigma^*$, $(\rho -+) \circ (\sigma + -) = (\sigma + \rho) -+ = \sigma\rho -+$. That is, for each $\omega \in \Sigma^*$, $\sigma + (\rho + \omega) = (\sigma + \rho) + \omega = \sigma\rho\omega$, and for each $x \in D$, $\sigma + (x + \rho) = (\sigma + \rho) + x = \sigma\rho + x$. Dually, for $\theta, \psi \in \Sigma^*$, $(-+\theta) \circ (-+\psi) = -+((\theta + \psi)) = -+\theta\psi$.

Using the endofunctors, similarity is now defined in terms of a set of generating ordered pairs.

**Definition 17.** Let $D_1 \subseteq D$ be the set of all derivations of length 1, $D_1 = \{x \mid x \in D \& l(x) = 1\}$. Let $\sim$ be a relation on $D$, $\sim \subseteq D^2$, defined as

$$\sim = \{(((x + \rho) \circ (\sigma + y), (\theta + y) \circ (x + \psi)) \mid x: \theta \to \sigma, y: \rho \to \psi \in D_1\}.$$ 

Diagrammatically, $(x + \rho) \circ (\sigma + y) \sim (\theta + y) \circ (x + \psi)$ is

$$\begin{array}{c}
\theta + \rho \\
\sigma + \rho \\
\sigma + \psi
\end{array} \sim
\begin{array}{c}
\theta + \rho \\
\theta + \psi \\
\sigma + \psi
\end{array}$$

Note that if $u \sim v$ then $d(u) = d(v)$ and $c(u) = c(v)$.

**Definition 18.** Similarity is the least congruence on $D = (\Sigma^*, D, d, c, \circ)$ containing the set $\sim$. We write $x \sim y$ just in case $x$ is similar to $y$. 

Lemma 19. For all derivations \( x : \theta \to \sigma, \ y : \rho \to \psi, \ (x + \rho) \circ (\sigma + y) \sim (\theta + y) \circ (x + \psi) \).

Proof. By induction on length. If \( l(x) = 0 \), then \( \theta = \sigma \) and \( x = 1_\theta \). In this case, \((1_\theta + \rho) \circ (\theta + y) = 1_{1_\theta} \circ (\theta + y) = \theta + y = (\theta + y) \circ 1_{1_\theta} = (\theta + y) \circ (1_\rho + \psi)\). Assume that for all derivations \( x : \theta \to \sigma \) of length \( n \) and for all derivations \( y : \rho \to \psi \) that \((x + \rho) \circ (\sigma + y) \sim (\theta + y) \circ (x + \psi)\). Now consider any derivation \( z : \theta \to \tau \) of length \( n + 1 \). There exist derivations \( x : \theta \to \sigma, \ w : \sigma \to \tau \) with \( l(x) = n \), \( l(w) = 1 \), and \( z = x \circ w \). Further, let \( y : \rho \to \psi \) be any derivation in \( D \). Now \((w + \rho) \circ (\tau + y) \sim (\sigma + y) \circ (w + \psi)\) by a second induction on the length of \( y \). Then compute

\[
(x + \rho) \circ (\sigma + y) = ((x \circ w) + \rho) \circ (\tau + y)
\]

\[
= (x + \rho) \circ (w + \rho) \circ (\tau + y)
\]

\[
\sim (x + \rho) \circ (\sigma + y) \circ (w + \psi)
\]

\[
\sim (\theta + y) \circ (x + \psi) \circ (w + \psi)
\]

\[
= (\theta + y) \circ ((x \circ w) + \psi)
\]

\[
= (\theta + y) \circ (x + \psi).
\]

With this lemma we obtain the following alternative for the definition of similarity.

Proposition 20. Similarity is the least congruence such that for all derivations \( x : \theta \to \sigma, \ y : \rho \to \psi \) in \( D \),

\[
(x + \rho) \circ (\sigma + y) \sim (\theta + y) \circ (x + \psi).
\]

Proof. Given the previous lemma, we have that

\[
\sim = \{((x + \rho) \circ (\sigma + y), (\theta + y) \circ (x + \psi)) \mid x : \theta \to \sigma, \ y : \rho \to \psi \in D \}
\]

is a subset of \( \sim \). But \( \sim \subseteq \sim \). Therefore the least congruence containing \( \sim \) is the similarity relation. \( \square \)

5. Syntax Categories

Let \( D/\sim \) denote the family of similarity equivalence classes. Each member of \( D/\sim \) is called a similarity class and is denoted by \([x]\) for some representative derivation \( x \). The quotient category of derivations modulo similarity, to be called the syntax category, is \( S = (\Sigma^*, D/\sim, \circ) \) where \( d : D/\sim \to \).
\[ \Sigma^* : [x] \mapsto d(x), \quad c : D/\sim \to \Sigma^* : [x] \mapsto c(x), \quad \text{and} \quad [x] \circ [y] = [x \circ y] \text{ for each} x, y \in D \text{ with } c(x) = d(y). \]

The \( x \)-category structure of \( S \) is now developed. The endofunctors \( \sigma + - : S \to S \) and \( - + \rho : S \to S \) are inherited from \( D \) by setting
\[
\sigma + - (\omega) = \sigma + \omega = \sigma \omega,
\]
\[
\sigma + - ([x]) = [\sigma + x], \quad \text{and}
\]
\[
- + \rho(\omega) = \omega + \rho = \omega \rho,
\]
\[
- + \rho([x]) = [x + \rho].
\]

As previously for derivations, we write \( \sigma + [x] \) for \( \sigma + - ([x]) \) and \( [x] + \rho \) for \( - + \rho([x]) \). That these functions are functors is seen by the calculation
\[
\sigma + [x \circ y] = [\sigma + (x \circ y)] = [(\sigma + x) \circ (\sigma + y)]
\]
\[
= [\sigma + x] \circ [\sigma + y] = (\sigma + [x]) \circ (\sigma + [y]),
\]
and equivalently for the functors \( - + \rho \). Note that again \( \sigma + ([x] + \rho) = (\sigma + [x]) + \rho \).

These endofunctors are used to define the second operation in the syntax category.

**Proposition 21.** Let \( x : \theta \to \sigma, \ y : \rho \to \psi \) be derivations in \( D \). Denote their images under the similarity epifunctor by \( [x] \) and \( [y] \), respectively. Then
\[
([x] + \rho) \circ (\sigma + [y]) = (\theta + [y]) \circ ([x] + \psi), \quad \text{that is, the following diagram}
\]
\[
\begin{array}{ccc}
\theta + [y] & \to & \theta + \psi \\
\sigma + [x] \searrow & & \nearrow \sigma + [y] \\
[x] + \rho & \to & [x] + \psi
\end{array}
\]

**Proof.** From Proposition 20, \( ([x] + \rho) \circ (\sigma + [y]) = (\theta + [y]) \circ ([x] + \psi) \).

Then calculating,
\[
([x] + \rho) \circ (\sigma + [y]) = [x + \rho] \circ [\sigma + y] = [(x + \rho) \circ (\sigma + y)] = [\theta + y] \circ [x + \psi] = (\theta + [y]) \circ ([x] + \psi). \quad \square
\]

**Definition 22.** Let the function \( - + - : (D/\sim)^2 \to D/\sim \) be given by the following data. For each pair \([x] : \theta \to \sigma, [y] : \rho \to \psi\), of similarity classes in \( S \), set
\[
[x] + [y] = ([x] + \rho) \circ (\sigma + [y]) = (\theta + [y]) \circ ([x] + \psi).
\]
The situation of the definition may not only be viewed as the diagonal of the previous commuting diagram, but also as

\[ [x] + [y] = [x] \begin{array}{c} \theta \vdash \rho \end{array} [y] \]

with \([x]\) and \([y]\) acting in parallel.

The concatenation operation on strings, \(+: (\Sigma^*)^2 \to \Sigma^*\), together with the above concatenation operation on similarity classes, \(+-: (D|\sim)^2 \to D|\sim\), gives a concatenation bifunctor on \(S\).

**Proposition 23.** The pair of functions \((+,-+-)\) is a bifunctor from \(S\) to \(S\), denoted by \(+: S^2 \to S\).

**Proof.** For each \(\sigma, \rho \in \Sigma^*\), \([1_\sigma] + [1_\rho] = ([1_\sigma] + \rho) \circ (\sigma + [1_\rho]) = [1_{\sigma+\rho}] = 1_{\sigma+\rho}\). For each \([x_1]: \sigma_0 \to \sigma_1\), \([x_2]: \sigma_1 \to \sigma_2\), \([y_1]: \rho_0 \to \rho_1\), \([y_2]: \rho_1 \to \rho_2\), in \(D|\sim\) calculate

\[ (\sigma_1 \circ \sigma_2) \circ (\rho_1 \circ \rho_2) = (\sigma_1 \circ \rho_0) \circ (\sigma_2 \circ (\rho_1 \circ \rho_2)) = (\sigma_1 \circ (\rho_1 \circ \rho_2)) \circ (\sigma_2 \circ (\rho_1 \circ \rho_2)) = (\sigma_1 \circ \rho_1) \circ (\sigma_2 \circ \rho_2) \]

The subsequent development in this section is entirely within the syntax category \(S\). To simplify the notation, write \(x \in D|\sim\), or \(x: \theta \to \sigma\) in \(D|\sim\), rather than continue to use the notation for similarity classes, \([x]\).

The proof of the above proposition establishes one of the conditions for a 2-category, that \((x_1 \circ x_2) + (y_1 \circ y_2) = (x_1 + y_1) \circ (x_2 + y_2)\) whenever both expressions are defined. The conditions on the identities are considered after showing that \(+: S^2 \to S\) is an associative bifunctor.

**Proposition 24.** For all \(x: \theta \to \sigma, y: \rho \to \psi, z: \tau \to \omega\) in \(D|\sim\), \((x+y)+z = x + (y+z)\).

**Proof.**

\[
(x + y) + z = ((x + y) + \tau) \circ (\sigma \psi + z) = (((x + \rho) \circ (\sigma + y)) + \tau) \circ (\sigma \psi + z) = ((x + \rho + \tau) \circ (\sigma + y + \tau) \circ (\sigma \psi + z) = (x + \rho \tau) \circ (\sigma + (y + \tau)) \circ (\sigma + (\psi + z)) = (x + \rho \tau) \circ (\sigma + (y + \tau)) \circ (\psi + z)) = (x + \rho \tau) \circ (\sigma + (y + z)) = x + (y + z). \]
Corollary 25. \( (\{\lambda\}, D/\sim, d', c', +) \) is a category, where \( d'(x) = c'(x) = \lambda \). \( \square \)

The only identity morphism for \( + \) is \( 1_\lambda \). Note that the system \( (D/\sim, +, 1_\lambda) \) is a monoid and that \( (\{\lambda\}, \{1_\lambda\}, d, c, o) \) is a subcategory of \( S \). Let \( \text{Id} = \{1_\theta | 1_\theta \in D/\sim, \theta \in \Sigma^*\} \). Since \( 1_\theta + 1_\psi = 1_{\theta + \psi} = 1_{\theta \phi} \), \( (\{\lambda\}, \text{Id}, d', c', +) \) is a subcategory of \( (\{\lambda\}, D/\sim, d', c', +) \). Since \( \lambda \in \Sigma^* \) and \( 1_\lambda \in \text{Id} \), the syntax category is the \( x \)-category \( S = (\Sigma^*, \{\lambda\}, D/\sim, d, c, o, d', c', +) \). In fact, \( S \) is a free \( x \)-category (Hotz, 1966).

Another way of viewing this information is by considering a certain collection of natural transformations between the endofunctors of the form \( \theta + \_ \). For each \( x: \theta \rightarrow \sigma, x \in D/\sim \), there is a natural transformation from the endofunctor \( \theta + \_ \) to the endofunctor \( \sigma + \_ \), which we will denote \( x_+ \). A natural transformation is a function from objects to morphisms: For each \( \theta \), let \( x_+ = x_\theta \). Now since for each \( y: p \rightarrow \psi \in D/\sim \),

\[
\begin{array}{ccc}
\theta + p & \xrightarrow{x + p} & \sigma + p \\
\theta + y \downarrow & & \downarrow \sigma + y \\
\theta + \psi & \xrightarrow{x + \psi} & \sigma + \psi
\end{array}
\]

commutes, \( x + : \theta + \rightarrow \sigma + \) is a natural transformation for each \( x: \theta \rightarrow \sigma \in D/\sim \). Then \( x + y \) is the diagonal of the above commuting diagram, and so a bifunctor on \( S \).

With the algebra available in \( S \), one may denote derivations by algebraic expressions involving \( o \) and \( + \). Each distinct well-formed algebraic expression denotes a derivation in \( D \) while the value of the algebraic expression is a similarity class in \( S \). Which is intended will be clear from usage. A derivation of length one applying the rewrite rule with index \( j \) in the context \( \mu \_ \nu \) is denoted \( (\mu + j + \nu) \). If \( P(j) = \alpha \rightarrow \beta \) then \( d(\mu + j + \nu) = \mu \alpha \nu \) and \( c(\mu + j + \nu) = \mu \beta \nu \). If \( j \) is the only index for \( \alpha \rightarrow \beta \) we may unambiguously write \( (\mu + (\alpha \rightarrow \beta) + \nu) \) or even \( (\mu(\alpha \rightarrow \beta) + \nu) \) for \( (\mu + j + \nu) \). The expressions for derivations of length one are called terms.

The expressions for derivations of length greater than one are obtained by writing compositions of terms

\[
(\mu_1 + j_1 + v_1) \circ (\mu_2 + j_2 + v_2) \circ \cdots \circ (\mu_n + j_n + v_n)
\]

for derivations of length \( n \). Each such composition of factors uniquely determines a derivation provided only that the composition is well-formed,
that is, that \( c(\mu_i + j_i + \nu_i) = d(\mu_{i+1} + j_{i+1} + \nu_{i+1}) \) for \( 1 \leq i < n \). The algebraic laws previously established may be used to form other expressions for the same similarity class in \( S \).

There is a considerable literature on the structure of syntax categories and relationships between syntax categories, e.g. (Claus, 1971; Claus and Walter, 1969; Hotz, 1966; Schnorr, 1969; Schnorr and Walter, 1969; Walter, 1970).

6. THE CATEGORY OF INTERCHANGE OPERATORS

The structure of similarity classes may be studied in terms of operators which carry a derivation into a similar one. In this section the interchange operators are defined and their basic algebraic properties considered. In Section 7 we demonstrate that two derivations are similar if and only if there is an interchange operator carrying one into the other.

The elementary interchange operators, to be defined, are partial functions from \( D \) to \( D \), where \( D \) is the set of all derivations in \((\Sigma, P)\). An elementary interchange operator applies at position \( j \) in a derivation if the productions applied at \( j \) and \( j + 1 \) do not overlap. If they do not overlap, the interchange operator switches the order of application of the two productions, resulting in a different, but equivalent, derivation.

**DEFINITION 26 (Griffiths, 1968).** The left elementary interchange operator \( L(j): D \rightarrow D \) is a partial function determined by the following data. \( x \in D \) is in the proper domain of the left interchange operator \( L(j) \) iff \( 0 < j < l(x) \), and \( l(\mu_j) \geq l(\mu_{j+1}) \), where \( P(r_{j+1}) = \alpha \rightarrow \beta \) and \( x = ((0_1, ..., 0_{j+1}), (r_1, ..., r_j), (\mu_1, ..., \mu_{j+1}, 0_{j+1}, ..., \mu_{n-1}, \nu_n)) \). If \( x \) is in the proper domain of \( L(j) \), it is the case that in \( x \)

\[
\begin{align*}
\theta_j &= \mu_{j+1} \alpha \phi \nu_j, \\
\theta_{j+1} &= \mu_{j+1} \alpha \phi \nu_{j+1}, \\
\theta_{j+2} &= \mu_{j+1} \beta \phi \nu_{j+2},
\end{align*}
\]

for some \( \phi \in \Sigma^* \), \( P(r_j) = \gamma \rightarrow \delta \), \( P(r_{j+1}) = \alpha \rightarrow \beta \), and so \( \mu_j = \mu_{j+1} \alpha \phi \), \( \nu_{j+1} = \phi \delta \nu_j \). The value of \( L(j) \) at \( x \) is written \( xL(j) \) and is the derivation \( xL(j) = ((0_1, ..., 0_j, \psi, \theta_{j+2}, ..., \theta_{n+1}), (r_1, ..., r_{j-1}, r_{j+1}, r_j, r_{j+2}, ..., r_n), (\mu_1, ..., \mu_{j-1}, \nu_{j-1}, \mu_{j+1}, \phi \gamma \nu_j, \mu_{j+1}, \beta \phi \nu_{j+2}, \mu_{j+2}, \nu_{j+2}, ..., \mu_{n-1}, \nu_n)) \), where \( \psi = \mu_{j+2} \beta \phi \nu_{j+2} \).

In \( xL(j) \) the production \( P(r_{j+1}) \), which is applied to the left of \( P(r_j) \), is done first. The left elementary interchange operators result in derivations in which production applications to the left occur before production applications
to the right. Using algebraic expressions, the subexpression of interest in $x$ is $(\mu_{j+1}\phi + r_j + \nu_j) \circ (\mu_{j+1} + r_{j+1} + \phi \nu_j)$ and the subexpression resulting from the application of $L(j)$ to $x$ is $(\mu_{j+1} + r_{j+1} + \phi \nu_j) \circ (\mu_{j+1}\phi + r_j + \nu_j)$.

We now define the dual situation of the right elementary interchange operators.

**Proposition 27.** Each $L(j): D \rightarrow D$ is injective. □

**Definition 28.** The right elementary interchange operator $R(j): D \rightarrow D$ is a partial function such that $x \in D$ is in the proper domain of $R(j)$ iff there is $y \in D$ such that $y L(j) = x$. In this case $x R(j) = y$.

The interchange operators are (i) the identity function on $D$, $I_D$, (ii) the left and right elementary interchange operators, and (iii) the functional compositions of interchange operators. In the usual way, $x T_1 T_2$ is defined for interchange operators $T_1$ and $T_2$ just in case $x T_1 = y$ is defined and $y T_2$ is defined. Note that if $T$ is an interchange operator applicable to $x \in D$, then $l(x T) = l(x)$, $d(x T) = d(x)$, and $c(x T) = c(x)$.

The following algebraic properties of interchange operators are from Griffiths (1968) and Langmaack (1971).

**Lemma 29.** (i) Let $\mid k - j \mid \geq 2$. If $T_1(j)$ and $T_2(k)$ are elementary interchange operators such that $T_1(j) T_2(k)$ applies to $x$, then $x T_1(j) T_2(k) = x T_2(k) T_1(j)$.

(ii) Let $\mid k - j \mid = 1$. If $L(j) R(k)$ applies at $x$ then $x L(j) R(k) = x R(k) R(j) L(k) L(j)$.

(iii) The dual of (ii) obtain by replacing $L$ by $R$ and $R$ by $L$ holds.

**Proof.** (Griffiths, 1968, Lemma 3.1). □

The situation in parts (ii) and (iii) of the lemma can be viewed as a commuting diagram. For simplicity, the diagram is specialized to the case of $k = 2$ and $j = 1$. The indices on the derivations show which rule applications have been interchanged. Only the left interchanges are drawn.
EXAMPLE 30. Let the rewriting system be that whose sole rewrite rule is $B \rightarrow c$. There are six derivations from $BBB$ to $ccc$, one for each ordering of rewriting the $B$'s to $c$'s. Let $x_{123}$ denote the derivation which rewrites the left $B$, then the middle one and finally the right $B$. The preceding diagram demonstrates the elementary interchanges possible among these six derivations.

EXAMPLE 31. The indexed rewriting system for this example has $Σ = \{A_1, A_2, B_1, B_2, B_3, C, D\}$, indexing set \{a, b, c, d\} and rewriting rules

\[
\begin{align*}
P(a) &= A_1 \rightarrow A_2, \\
P(b) &= B_1 \rightarrow B_2B_3, \\
P(c) &= B_3 \rightarrow C, \\
P(d) &= A_2B_2 \rightarrow D.
\end{align*}
\]

There are five derivations from $A_1B_1$ to $DC$ which we denote in a brief manner as follows.

1. $A_1B_1 \rightarrow A_2B_1 \rightarrow A_2B_2B_3 \rightarrow DB_3 \rightarrow DC$
2. $A_1B_1 \rightarrow A_2B_1 \rightarrow A_2B_2B_3 \rightarrow A_2B_2C \rightarrow DC$
3. $A_1B_1 \rightarrow A_1B_2B_3 \rightarrow A_2B_2B_3 \rightarrow A_2B_2C \rightarrow DC$
4. $A_1B_1 \rightarrow A_1B_2B_3 \rightarrow A_2B_2B_3 \rightarrow A_2B_2C \rightarrow DC$
5. $A_1B_1 \rightarrow A_1B_2B_3 \rightarrow A_1B_2C \rightarrow A_2B_2C \rightarrow DC$.

The derivations may be denoted in an even more compact fashion as paths from the top to the bottom of the following diagram.

The interchanges among these five derivations are indicated by the diagram.

\begin{center}
\includegraphics{diagram}
\end{center}

643/28/1-2
One sees that the hypotheses of Lemma 29 (ii) and (iii) do not hold.

Each interchange operator $T$ has an inverse as follows. The identity $I_D$ is its own inverse. $L(k)$ and $R(k)$ are inverses of each other. If $T$ is the functional composition of a sequence of elementary interchange operators, $T = T_1(j_1) \cdots T_n(j_n)$, where each $T_i$ is an elementary interchange operator, let $T_i^{-1}$ denote the inverse of $T_i$. Then $T_n^{-1}(j_n) \cdots T_1^{-1}(j_1)$ is the inverse of $T$. Note that $TT^{-1} \neq I_D$ in general, since $T$ and $T^{-1}$ are partial functions and $I_D$ is total.

When using interchange operators to explore the algebraic properties of derivations, the questions reduce to the existence of an interchange operator $T$ such that $xT = y$ for certain $x$ and $y$. This suggests forming a category with derivations as the objects and the triples $(x, T, xT)$ as the morphisms. This can be done, but the resulting category lacks certain useful properties. For example, the composition of $(x, T, y)$ and $(y, T^{-1}, x)$ does not result in the identity morphism $(x, I_D, x)$. To obtain the desired properties, a more complex construction is required.

Let $N = \{1, 2, \ldots\}$ be the set of natural numbers. Each elementary interchange operator $T(j)$ induces the transposition $(j \ j + 1)$ on $N$. By composition of the induced transpositions, each sequence of elementary interchange operators $(T_1(j_1), \ldots, T_n(j_n))$ induces a permutation

$$p = (j_1 \ j_1 + 1)(j_2 \ j_2 + 1) \cdots (j_n \ j_n + 1).$$

$p$ has the property that for $m = \max\{j_1, \ldots, j_n\}$ and all $k > m + 1$, $kp = k$. Clearly each sequence of elementary interchange operators determines an interchange operator by functional composition.

**Definition 32.** Two sequences of elementary interchange operators, $(T_1(j_1), \ldots, T_n(j_n))$ and $(T_1'(k_1), \ldots, T_m'(k_m))$, are equivalent with respect to $x$ and $y$ if the induced permutations are equal, $(j_1 \ j_1 + 1) \cdots (j_n \ j_n + 1) = (k_1 \ k_1 + 1) \cdots (k_m \ k_m + 1)$, and the interchange operators

$$T = T_1(j_1) \cdots T_n(j_n) \quad \text{and} \quad T' = T_1'(k_1) \cdots T_m'(k_m)$$

satisfy $xT = y = xT'$. Let $E(x, y)$ denote the equivalence classes so formed.

If $E(x, y)$ and $E(y, z)$ are such equivalence classes, then there is a unique equivalence class, denoted $E(x, y) \ast E(y, z)$, containing all sequences $(T_1(j_1), \ldots, T_n(j_n)), T_1'(k_1), \ldots, T_m'(k_m))$ for $(T_1(j_1), \ldots, T_n(j_n)) \in E(x, y)$ and $(T_1'(k_1), \ldots, T_m'(k_m)) \in E(y, z)$. 
Let $\mathcal{T}$ be the set of all triples $(x, E(x, y), y)$ for $x, y \in D$. The domain function $d: \mathcal{T} \to D$ is $d(x, E(x, y), y) = x$ and the codomain function function $c: \mathcal{T} \to D$ is $c(x, E(x, y), y) = y$. For each pair $(x, E(x, y), y)$ and $(y, E(y, z), z)$ in $\mathcal{T}$ the composition is $(x, E(x, y), y) \ast (y, E(y, z), z) = (x, E(x, y) \ast E(y, z), z)$. Since the null sequence induces the identity permutation and functionally determines $I_D$, let $I(x)$ be the equivalence class containing the null sequence. Then $(x, I(x), x)$ is the categorical identity on $x \in D$ for the category $T = (D, \mathcal{T}, d, c, \ast)$. $T$ is called the interchange category over $D$. Each morphism $(x, E(x, y), y)$ in $\mathcal{T}$ is called an interchange, not to be confused with an interchange operator.

Interchanges $(x, E(x, y), y)$ will be denoted $T: x \to y$ for some interchange operator $T$ determined by functional composition of a sequence in $E(x, y)$. This vastly simplified the subsequent presentation.

**Proposition 33.** $T: x \to y \ast T^{-1}: y \to x$ is the identity on $x$. Lemma 29 holds for interchanges.

**Proof.** Recalling the definition of inverses, the permutation induced by $E(x, y) \ast E(y, x)$ is the identity and $xTT^{-1} = x$. So the null sequence is in $E(x, y) \ast E(y, x)$. Lemma 29 holds for interchanges since transpositions on $N$ satisfy the properties that if $|k - j| \geq 2, (jj+1)(kk+1) = (kk+1)(jj+1)$ and if $|k - j| = 1, (jj+1)(kk+1)(jj+1) = (kk+1)(jj+1)(kk+1)$.

There is a second operation available in $T$, denoted $\circ$ since it is directly related to the composition of derivations. $T$ equipped with $\circ$ is a 2-category. The details are very similar to the development of the syntax category and are not given. The construction of $\circ$ is sketched in the remainder of the section.

Let $T(j): x \to y$ be an interchange in which $T(j)$ is an elementary interchange operator, and $x$ and $y$ are derivations of length $n$. Let $z$ be any derivation such that $c(x) = d(z)$. Then the elementary interchange operator $T(j)$ applies to $x \circ z$ and $T(j): x \circ z \to y \circ z$ is an interchange of $T$. Let $w$ be a derivation of length $k$ such that $c(w) = d(x)$. Since $T(j)$ is applicable to $x$ at position $j$ and interchange operators preserve length, $T(k + j)$ is applicable to $w \circ x$ at position $k + j$, so that $T(k + j): w \circ x \to w \circ y$ is an interchange of $T$. By iterating, one obtains that for each $T: x \to y$ in $T$ and each $z \in D$ such that $c(x) = d(z)$ there is an interchange denoted $T \circ z: x \circ z \to y \circ z$ moving $x$ to $y$ while leaving $z$ fixed. Dually, for each $w \in D$ such that $c(w) = d(x)$ there is an interchange denoted $w \circ T: w \circ x \to w \circ y$ leaving $w$ fixed. If the compositions are defined, $(w \circ (T \circ z)) = (w \circ T) \circ z) = w \circ T \circ z.$
Proposition 34. If $T_1: x_1 \to y_1$ and $T_2: x_2 \to y_2$ are interchanges of $T$ for which $c(x_1) = d(x_2)$ then the following diagram commutes. \[\]

\[
\begin{array}{c}
\text{x}_1 \xrightarrow{T_1} \text{x}_2 \\
\text{y}_1 \xrightarrow{T_2} \text{y}_2 \\
\end{array}
\]

Definition 35. For each pair $T_1: x_1 \to y_1$, $T_2: x_2 \to y_2$ of interchanges of $T$, let $T_1 \circ T_2 = (T_1 \circ x_2) \ast (y_1 \circ T_2) = (x_1 \circ T_2) \ast (T_1 \circ y_2)$ if $c(x_1) = d(x_2)$.

The situation of the definition may not only be viewed as the diagonal of the previous commuting diagram, but also as $T_1$ and $T_2$ acting in parallel.

7. Interchanges and Simplicity Classes

We demonstrate that two derivations, $x, y, in D$ are similar, $x \sim y$, if and only if there exists an interchange $T: x \to y$ in the 2-category of interchanges. This will complete the demonstration that the notions of similarity used in Griffiths (1968) and Benson (1970) coincide. An alternative proof is given in Langmaack (1967).

Lemma 36. Let $T$ be an elementary interchange operator. $T: x \to y$ is an interchange in $T$ iff there exists $w, \tilde{x}, \tilde{y}, z \in D$ such that $x \sim w \circ \tilde{x} \circ z$, $y = w \circ \tilde{y} \circ z$ and $\tilde{x} \sim \tilde{y}$ or $\tilde{y} \sim \tilde{x}$.

Proof. Suppose $T = L(j)$ and $xL(j) = y$. Then the length of $x$ is at least $j + 1$. Let $l(w) = j - 1$, $\tilde{x}$ be the $j$th and $j + 1$st components of $x$, and $z$ be the remainder of $x$ so that $x = w \circ \tilde{x} \circ z$. Let $\tilde{y} = \tilde{x}L(1)$. Then $y = w \circ \tilde{y} \circ z$ and by definition $\tilde{y} \sim \tilde{x}$. A similar argument shows that if $T = R(j)$, one obtains $\tilde{x} \sim \tilde{y}$. Now suppose $\tilde{y} \sim \tilde{x}$, $x = w \circ \tilde{x} \circ z$, and $y = w \circ \tilde{y} \circ z$. Then $\tilde{x}L(1) = \tilde{y}, w \circ L(1) \circ x : w \circ \tilde{x} \circ z \rightarrow w \circ \tilde{y} \circ z$ is an interchange in $T$ and $w \circ L(1) \circ z$ is clearly elementary. Dually if $\tilde{x} \sim \tilde{y}$. \[\]

Lemma 37. If $T: x \to y$ is an interchange in $T$ then $x \sim y$.\[\]
Proof. Given \( T: x \rightarrow y \), \( T = T_1(j_1) \cdots T_p(j_p) \), where each \( T_i(j_i) \) is an elementary interchange operator. Using the previous lemma in an immediate induction, one obtains that \( x \sim y \). \( \Box \)

The above is half of the desired result. For the other half, it is useful to introduce the relation \( \approx \) on derivations, defined by \( x \approx y \) iff there exists an interchange \( T: x \rightarrow y \). That is, \( x \approx y \) iff \( \text{Mor}_T(x, y) \neq \emptyset \).

**Lemma 38.** \( \sim \subseteq \approx \). Furthermore, \( \approx \) is a congruence relation on the category \( D = (\Sigma^*, D, d, c, o) \).

Proof. By Lemma 36, \( \sim \subseteq \approx \). Now \( \approx \) is an equivalence relation since there are identity interchanges, inverse interchanges, and if \( T_1: x \rightarrow y \), \( T_2: y \rightarrow z \) then \( T_1 \ast T_2: x \rightarrow z \). Since interchanges preserve the domains and codomains of the interchanged derivations, the domain and codomain requirements of a congruence are met. Finally, if \( T_1: x_1 \rightarrow y_1 \) and \( T_2: x_2 \rightarrow y_2 \) then \( T_1 \circ T_2: x_1 \circ x_2 \rightarrow y_1 \circ y_2 \) so that the compositional requirement is met. \( \Box \)

**Proposition 39.** \( x \sim y \) iff there exists an \( T: x \rightarrow y \), an interchange of \( T \).

Proof. It remains only to show that if \( x \sim y \) then \( T: x \rightarrow y \) exists. By the previous lemmas, \( \approx \) is a congruence such that \( \sim \subseteq \approx \subseteq \approx \). Since similarity is the least congruence, \( \approx = \sim \). That is to say, if \( x \sim y \) then there exists a \( T: x \rightarrow y \). \( \Box \)

**Corollary 40.** If \( x \sim y \) then \( l(x) = l(y) \). \( \Box \)

By the above proposition, \( T \) is the union of full subcategories \( T[x] \), each subcategory having a similarity class as its set of objects.

### 8. Interchange Groups on Similarity Classes

For each similarity class \( [x] \), the elementary interchange operators \( T(j): D \rightarrow D \) can be viewed as restricted to \( [x] \), \( T(j): [x] \rightarrow [x] \). These restrictions together with the additional fact of representability give data specifying bijections permuting the derivations in \( [x] \). We first define and characterize representability and then consider the bijections of interest.

**Definition 41.** \( [x] \) is not representable if for some \( y \in [x] \), the \( j \)th and \( j+1 \)st terms of \( y \) are \( (\mu + r_j + \nu) \circ (\mu + r_{j+1} + \nu) \), where \( P(r_j) = \alpha \rightarrow \lambda \), \( P(r_{j+1}) = \lambda \rightarrow \beta \) and \( \alpha \beta \neq \lambda \). Otherwise \( [x] \) is representable.
The results in this section show that a representable similarity class may be represented by a permutation group of operators acting on the derivations of the class. The results in the next section show that if all similarity classes in a syntax category are representable, then each similarity class may be represented by a certain ring of sets.

The following gives the conditions under which a similarity class is representable. Recall that $\lambda$ denotes the null string in $\Sigma^*$.

**Theorem 42.** Let $L(j)$ be a left elementary interchange operator. $L(j)$ and $R(j)$ are both applicable to $y$ iff the $j$th and $j + 1$st terms of $y$ are

$$(\mu + r_{j + 1} + v) \circ (\mu + r_{j + 1} + v),$$

where $P(r_j) = \alpha \rightarrow \lambda, P(r_{j + 1}) = \lambda \rightarrow \beta, \alpha, \beta, \mu, v \in \Sigma^*$. Further, $yL(j) = yR(j)$ iff $\alpha \beta = \lambda$.

**Proof.** Let $y = ((\theta_1, ..., \theta_{n+1}), (r_1, ..., r_n), (\mu_1, v_1, ..., \mu_n, v_n))$. Since $L(j)$ is applicable to $y$, the following equations hold.

$$\theta_j = \mu_{j+1} \omega \phi \omega j,$$

$$\theta_{j+1} = \mu_{j+1} \omega \phi \delta v,$$

$$\theta_{j+2} = \mu_{j+1} \beta \delta \phi \delta v,$$

with $P(r_j) = \alpha \rightarrow \delta, P(r_{j + 1}) = \omega \rightarrow \beta, \mu_j = \mu_{j+1} \omega \phi, v_{j + 1} = \phi \delta v_j$ for some $\phi$.

Since $R(j)$ also applies to $y$,

$$\theta_j = \mu_{j+1} \xi \omega \omega j,$$

$$\theta_{j+1} = \mu_{j+1} \xi \omega \delta v,$$

$$\theta_{j+2} = \mu_{j+1} \xi \beta \phi \delta v,$$

and $\mu_{j+1} = \mu \xi \phi, v_j = \omega \omega \omega j$ for some $\xi$. Solving these equations gives

$$\delta = \omega = \phi = \xi = \lambda, \quad \mu_j = \mu_{j+1}, \quad v_j = v_{j + 1}.$$

That is, step $j$ in $y$ is $((\mu \omega v, \mu v), (r_j), (\mu v))$ and step $j + 1$ in $y$ is $((\mu v, \mu \beta v), (r_{j + 1}), (\beta v))$ with $P(r_j) = \alpha \rightarrow \lambda, P(r_{j + 1}) = \lambda \rightarrow \beta, \mu = \mu_j, v = v_j$.

To prove the second assertion of the theorem, let step $j$ and step $j + 1$ in $y$ be

$$((\mu \omega v, \mu v, \mu \beta v), (r_j, r_{j + 1}), (\mu v, \mu v))$$

with $P(r_j) = \alpha \rightarrow \lambda, P(r_{j + 1}) = \lambda \rightarrow \beta$. Then steps $j$ and $j + 1$ of $yL(j)$ are

$$((\mu \omega v, \mu \beta v, \mu \beta v), (r_j, r_{j + 1}), (\mu v, \mu \beta v))$$
while steps $j$ and $j+1$ of $yR(j)$ are
\[(\mu X, \mu \alpha \beta \nu, \mu \beta \nu, (r_{j+1}, r_j), (\mu \alpha, \mu \beta \nu)).\]
These last two derivations differ iff $\alpha \beta \neq \lambda$. 

The situation of the theorem may be viewed as follows. Let $(\Sigma, P)$ be such that $P(1) = \alpha \rightarrow \lambda$, $P(2) = \lambda \rightarrow \beta$, where not both $\alpha$ and $\beta$ are $\lambda$, i.e., $\alpha \beta \neq \lambda$. The following three derivations, which we write in both sequence and algebraic form, form a similarity class.

\[r = ((\alpha, \alpha \beta, \beta), (2, 1), (\alpha \lambda, \lambda \beta)) = (\alpha + (\lambda \rightarrow \beta)) \circ ((\lambda \rightarrow \lambda) + \beta),\]
\[m = ((\alpha, \lambda, \beta), (1, 2), (\lambda \alpha, \lambda \lambda)) = (\lambda \rightarrow \lambda) \circ (\lambda \rightarrow \beta),\]
\[l = ((\alpha, \beta \alpha, \beta), (2, 1), (\alpha \alpha, \alpha \lambda)) = ((\lambda \rightarrow \beta) + \alpha) \circ (\beta + (\alpha \rightarrow \lambda)).\]

The graph of the left and right interchanges of the interchange category is

\[
\begin{array}{ccc}
L(I) & m & R(I) \\
\downarrow & \downarrow & \downarrow \\
L(I) & m & R(I)
\end{array}
\]

The graph of the three derivations as paths is

\[
\begin{array}{ccc}
\beta & 2 & \beta \\
\alpha & \lambda & \beta \\
\lambda & \beta & \lambda \\
\end{array}
\]

with $l$ on the left, $m$ in the middle, and $r$ on the right.

If $P(1) = P(2) = \lambda \rightarrow \lambda$ then the derivations $l$ and $r$ collapse to $((\lambda, \lambda, \lambda), (2, 1), (\lambda \lambda, \lambda \lambda))$ and the resulting path diagram becomes

\[
\begin{array}{ccc}
1 & 2 & 1 \\
\lambda & \lambda & \lambda \\
\end{array}
\]

Corollary 43. Let $[x]$ be representable. For each elementary interchange operator $T(j): [x] \rightarrow [x]$ let $g_j: [x] \rightarrow [x]$ be determined by

\[y g_j = \begin{cases} 
  y T(j) & \text{if } T(j) \text{ applies to } y, \\
  y T^{-1}(j) & \text{if } T^{-1}(j) \text{ applies to } y, \\
  y & \text{otherwise.}
\end{cases}\]

Then $g_j$ is a function, in fact a bijection.
We now consider the interchange group. For the remainder of the section let \([x]\) be a representable similarity class.

Let \(G[x]\) denote the set formed from the \(g_i; \ [x] \rightarrow [x]\) by closure under functional composition. By the previous corollary \(G[x]\) is a group of transformations, in the usual sense, such that each \(g \in G[x]\) is its own inverse. \(G[x]\) will be called the interchange group on the representable similarity class \([x]\).

It is easy to see that if \(|k - j| \geq 2\) then \(g_jg_k = g_kg_j\). But if \(|k - j| = 1\), \(g_jg_kg_j\) is not, in general, equal to \(g_kg_jg_k\). Therefore in general the group \(G[x]\) is not the braid group (Garside, 1969). To see this last, refer to Example 45.

If \(T: y \rightarrow z\) is a interchange of the interchange category with \(y, z \in [x]\) such that \(T = T_1(j_1) T_2(j_2) \cdots T_x(j_x)\) with \(T_x(j_x)\) elementary then \(y, g_jg_k \cdots g_j = z\), but Example 45 shows there is in general no functor from the interchange subcategory \(T[x]\) to \(G[x]\).

**Example 44.** The rewriting system is that of Example 30. The similarity class is that of all derivations from \(BBB\) to \(ccc\). Using the notation of Example 30, the derivations are \(x_{123}, x_{213} \ldots, x_{921}\). Using cycle notation, the generators of this interchange group are

\[
g_1 = (x_{123} x_{213})(x_{231} x_{321})(x_{132} x_{312})
g_2 = (x_{123} x_{132})(x_{213} x_{321})(x_{312} x_{231})
\]

The interchange group is a permutation group of order 6 on 6 letters and is isomorphic to the symmetric group \(S_6\). The elements of the group are \(e, g_1, g_2, g_1g_2, g_2g_1, \) and \(g_1g_2g_1 = g_2g_1g_2\).

**Example 45.** In Example 31, the derivations are denoted by 1, 2, 3, 4, 5. The generators for this interchange group are

\[
g_1 = (1 3)(2 4)\]
\[
g_2 = (4 5)\]
\[
g_3 = (1 2)(3 4).
\]

The interchange group generated is the symmetric group on 5 letters, \(S_5\). Note that \(g_1g_2g_3 \neq g_2g_1g_2\).

Since each derivation in a similarity class, \([x]\), is carried into each other derivation in the class by interchange operators, \(G[x]\) is a transitive permutation group. If every similarity class in \(D/\sim\) is representable, then there is an
intransitive interchange group $G(D)$ permuting the derivations of $D$. The
definition of $G(D)$ may be obtained in analogy to Corollary 43 by considering
the elementary interchange operators to act on all of $D$. The sets of transitivity
for $G(D)$ are exactly the similarity classes.

The interchange groups characterize the situation in which it is not possible
to tell whether a left or a right interchange is being applied, only that some
permutation of the order of rule application occurs. These groups are closely
related to the partial orders considered in the next section.

9. Partial Orders in Representable Similarity Classes

In this section, we consider left-divisibility of derivations. The principal
result is that uniform representability of the indexed rewriting system
implies that the set of left divisors of each similarity class is a distributive
lattice. The careful proof we give requires considerable apparatus, much of
it of independent interest.

Definition 46. An indexed rewriting system $(\Sigma, P)$ is uniformly
representable if $P(j) \in \Sigma^* \times \Sigma^*$ for all $j \in J$ and uniformly corepresentable if
$P(j) \in \Sigma^* \times \Sigma^+$ for all $j \in J$.

Definition 47. Let \([x], [z]\) be morphisms of $S$. \([x]\) divides \([z]\), \([x]\mid [z]\),
if there exists \([y]\) such that \([x] \circ [y] = [z]\). Let $\mathcal{L}_{[x]}$ be the set of divisors
of \([x]\).

Clearly $(\mathcal{L}_{[x]}, \mid)$ is a poset (partially ordered set). We will establish that if
$(\Sigma, P)$ is uniformly representable then for all \([x]\), $(\mathcal{L}_{[x]}, \mid)$ is a distributive
lattice. The proof proceeds as follows: By the use of the partial order of rule
application dependence, a certain ring of sets is established. Then certain
properties of uniform representability are derived, and these properties used
to establish an order-isomorphism between the ring of sets and $\mathcal{L}_{[x]}$. This
will complete the proof.

The following intuition motivates the technical definition of rule application
dependency. Given a derivation $x$, one rule application of $x$ depends on
another if the domain of the first rewriting rule requires, perhaps indirectly,
some portion of the codomain string resulting from the depending rewrite
rule. Since the same rewrite rule with a given index may be applied more than
once in a given derivation, it is rule applications which depend on one another,
not just the rewrite rules.
Rule application dependency can be defined via graph theoretic techniques following Griffiths (1968) or Hotz (1966). Here it is most convenient to define the dependency in terms of rule application sequences and interchanges.

**Definition 48.** For each similarity class \([x]\) fix a representative \(x\). Let \(n = l(x)\). Fix a bijection \(b: n \to A\), where \(A \subseteq N\) and \(#(A) = n\). Usually, \(A = n\) and \(b\) is the identity function, but other cases will arise. Let \(T: x \to y\) be an interchange inducing the permutation \(p\). Let \(p'\) be the permutation of \(A\) such that \(p' = b^{-1} \circ (p \mid n) \circ b\). \(p'\) is a rule application sequence for \(y\) with respect to \(x\) and \(b\).

In general a rule application sequence of \(y\) with respect to \(x\) is not unique. For example, if \(P(a) = \lambda \to \lambda\), the morphism \((\lambda \to a \lambda) \circ (\lambda \to a \lambda)\) in \(S\) has a unique representative but the two rule application sequences \((1, 2)\) and \((2, 1)\).

For a rule application sequence \(p' = (i_1, i_2, \ldots, i_n)\) say that \(j\) precedes \(k\) if \(j = i_s\), \(k = i_t\) for some \(s < t\), \(j\) immediately precedes \(k\) at \(s\) if \(t = s + 1\), and \(j\) is adjacent to \(k\) if one of \(j, k\) immediately precedes the other.

**Definition 49.** Rule application dependency with respect to \(x\) and \(b\) is denoted \(\leq(x, b)\) and is defined on \(A\). Let \(T: x \to z\) induce the permutation \(p_T\). Define \(jb \leq (T, b) \iff j p_T \leq k p_T\). Then

\[
\leq(x, b) = \bigcap \{\leq(T, b) \mid T: x \to z\text{ for some }z\},
\]

Clearly \(\leq(x, b)\) is a partial order. If \(b\) is the identity \(\leq(x, b)\) is denoted \(\leq_x\).

Note that \(\leq_x \subseteq \leq\).

Referring to Example 31, the rule sequence for the canonical derivation is \((a, b, d, c)\) and the rule application sequence is \((1, 2, 3, 4)\). The covering graph for rule application dependency is

\[
\begin{array}{c}
3 \\
\downarrow \\
1 \\
\uparrow \\
4
\end{array}
\]

If the rewriting system is context free, the covering graph is a multiple-rooted tree.

If \((Q, \leq)\) is a poset, setting \(B_Q = \{X \subseteq Q \mid x \in X \implies y \in X\} \subseteq Q\) results in a ring of sets \((B_Q, \subseteq)\), "the sets below" (MacLane and Birkhoff, 1967). For each \((n, \leq_n)\) derived as above from a similarity class \([x]\), let \((B_x, \subseteq)\) be the ring of sets below, a distributive lattice. Each \(X \in B_x\) roughly represents a divisor of \([x]\), \(\emptyset \in B_x\) represents \(1_{\mathcal{Q}(x)}\), \(n \in B_x\) represents \([x]\). The representation is exact in the case of uniformly representable indexed
rewriting systems, as is shown after deriving certain consequences of uniform representability.

If a left elementary interchange $L(j)$ applies to $y \in [x]$, write $yL(j) \lessdot y$. The transitive closure of $\lessdot$ is also denoted $\cdot$. If $z \lessdot y$, we say that $z$ is more canonical than $y$, or that $z$ is to the left of $y$. If $(\Sigma, P)$ is uniformly representable, the reflexive closure of $\lessdot$ is a partial order, in fact a modular lattice (Langmaack 1971). The least element under $\lessdot$ is called the canonical, or left, derivation in $[x]$ and the greatest element is called the right derivation in $[x]$. The general conditions for the existence of canonical derivations are given in Griffiths (1968). Langmaack (1971) contains a thorough study of the relation of "more canonical than," in particular, its connection with the braid group.

**Lemma 50.** If $x$ is canonical, then for each $y \in [x]$ there is a unique $T: y \rightarrow x$.

**Proof.** Assume $(y, E_1, x)$ and $(y, E_2, x)$ are interchanges. By Griffiths' Theorem 3.1 there is a sequence $(L(i_1),...,L(i_n))$ in $E_1$ and $(L(j_1),...,L(j_n))$ in $E_2$. If $E_k$ induces the permutation $p_k$, $k = 1, 2$, then

$$H = (R(j_n),...,R(j_1),L(i_1),...,L(i_n))$$

takes $x$ to $x$ and induces the permutation $p_2^{-1}p_1$. Again following Griffiths' Theorem 3.1, the null sequence is equivalent to $H$ hence $p_2^{-1}p_1$ is the identity permutation so $E_1 = E_2$. 

**Corollary 51.** If $x$ is canonical, there is a unique interchange $T: y \rightarrow z$ for any $y, z \in [x]$. Further $T: x \rightarrow y$ contains a sequence of right elementary interchange operators $(R(j_1),...,R(j_n))$. 

The set of all rule application sequences for $[x]$ is complete in the sense that each permutation compatible with $\preceq$ is a rule application sequence as is shown in the next theorem.

**Theorem 52.** Let $x$ be canonical, $n = l(x)$. Let $p = (i_1, i_2, ..., i_n)$ be any permutation of $n$ which is compatible with $\preceq$. That is, $j < k$ implies $i_k \preceq i_j$. Then there is a derivation $y \in [x]$ such that $y$ has $p$ as a rule application sequence with respect to $x$ and the identity function on $n$.

**Proof.** The existence of an interchange $T: x \rightarrow y$ with $p_T = p$ will suffice. In $p = (i_1, i_2, ..., i_n)$, $i_1$ is a minimal element in the $\preceq$-partial order. By Corollary 51 $R(i_1 - 1)R(i_1 - 2) \cdots R(1)$ is applicable to $x$ inducing the
permutation \((i_1, 1, 2, ..., i_1 - 1, i_1 + 1, ..., n)\). By induction, assume \(T_1\) is
applicable to \(x\) inducing the permutation \((i_1, i_2, ..., i_u, j_{u+1}, ..., j_n)\). \(i_{u+1}\)
depends on at most \(i_1, i_2, ..., i_u\) since \(p\) is compatible with \(\leq_x\). Let \(i_{u+1} = j_s\).
Then \(T_1 R(j_s-1) \cdots R(u+1)\) is applicable to \(x\) and induces the permutation
\((i_1, i_2, ..., i_u, i_{u+1}, j_{u+1}, ..., j_{s-1}, j_{s+1}, ..., j_n)\). □

An alternative approach to defining rule application dependency, \(\leq_x\), in
representable \([x]\) is to use the interchange group to induce the permutations.
The method above, using interchanges, is slightly more satisfactory. The
next lemma establishes the consistency of rule application dependency.

**Lemma 53.** Let \(x\) be canonical, \([x] = [z] \circ [u]\) and \(z\) canonical in \([z]\) with
\(k = l(z)\). For \(T: x \rightarrow z \circ u\) define \(b = p_T^{-1} | k\). Then \(\leq(z, b) = \leq_x | \text{Im}(b)^2\).

**Proof.** For each \(T: x \rightarrow y\), \(p_T\) is an order-isomorphism from \((n, \leq_{x})\) to
\((n, \leq_{y})\) and \(\leq_x = \leq(y, p_T^{-1} | n)\). Each \(u\) for which \(T: x \rightarrow z \circ u\) exists
establishes the equalities,

\[
\leq(z, b) = \leq(z \circ u, p_T^{-1} | n) | \text{Im}(b)^2 = \leq_x | \text{Im}(b)^2. \quad \square
\]

With the foregoing we demonstrate that there exists an order-preserving
function \(f: B_x \rightarrow \mathcal{L}_{[x]}\), for canonical \(x\). Let \(n = l(x)\) and \(f(n) = [x]\). By
induction assume that for all \(X \subseteq B_x\) with \(#(X) = k > 0\) that \(f(X) \in \mathcal{L}_{[x]}\)
and \(X \subseteq Z\) implies \(f(X) | f(Z)\). Consider any \(Y \in B_x\) with \(#(Y) = k - 1\).
Let \(j\) be the numerically smallest integer in \(n\) such that \(j \notin Y\). \(Y' = Y \cup \{j\} \subseteq B_x\) since every \(i \leq_x j\) is in \(Y\). Let \(z\) be the canonical representative
of \(f(Y')\). There exists \(u, T\) such that \(T: x \rightarrow z \circ u\) by divisibility. Then by
Lemma 53 and Theorem 52 there exist \(\mu_1, \nu_1, 1 \leq s \leq k\), such that \(f(Y') =
(\mu_1 + r_1^x + \nu_1) \circ \cdots \circ (\mu_{k-1} + r_1^x + \nu_{k-1}) \circ (\mu_k + r_x^z + \nu_k)\),
where each \(i_s \in Y\), \(1 \leq s < k\), and \((i_1, ..., i_{k-1}, j)\) is a rule application sequence with respect to \(z\)
and \(p_T^{-1} | k\). Define \(f(Y) = o_{s=1}^{k-1} (\mu_s + r_s^x + \nu_s)\). Clearly \(f(Y) | f(Y')\) so
\(f(Y) \in \mathcal{L}_{[x]}\) and \(f\) is order-preserving. Note that \(f(\emptyset) = 1_{d(a)}\).

The characterization function \(g: \mathcal{L}_{[x]} \rightarrow B_x\) for canonical \(x\) is \(g[y] =
\text{Im}(p_T^{-1} | k)\), where \(T: x \rightarrow y \circ z\) and \(k = l(y)\). Note that \(g[x] = n\) and
\(g[1_{d(a)}] = \emptyset\). The definition is independent of the choice of representative,
since for \(y' \in [y]\), \(z' \in [z]\) there is a unique \(T': x \rightarrow y' \circ z'\). Further, \(T': y \rightarrow y'\)
and \(T'': z \rightarrow z'\) are unique. Then \(T' = T' \circ (T'' \circ T'')\) and \(\text{Im}(p_T^{-1} | k) =
\text{Im}(p_T^{-1} | k)\). Clearly \(g\) is order-preserving, \(g\) is a surjection by the existence of
\(f: B_x \rightarrow \mathcal{L}_{[x]}\) for we have \(g(f(Y)) = Y\).

We will require the following cancellation property of uniformly
representable indexed rewriting systems.
Proposition 54. If \((\Sigma, P)\) is uniformly representable then every morphism in \(S\) is epic.


By duality, if \((\Sigma, P)\) is uniformly corepresentable then every morphism in \(S\) is monic. However, these are the strongest conclusions possible, since \(\Sigma = \{a\}, P(1) = a \to \lambda\), is uniformly representable while for \(x_1 = (a \to \lambda) + a, x_r = a + (a \to \lambda), y = a \to \lambda, x_1 \circ y = x_r \circ y\) but \(x_1 \neq x_r\) so \(x_1, x_r\) are not monic.

Lemma 55. If \((\Sigma, P)\) is uniformly representable then \(g: \mathcal{L}_x \to B_x\) is an injection for each canonical \(x\).

Proof. \(g(x) = \emptyset\) if \(x = 1_{d(x)}\). The induction hypothesis is that for each \(Y \in B_x\) with \(#(Y) = k\) there is a unique \([y] \in \mathcal{L}_{[x]}\) such that \(g[y] = Y\). Consider any \(X \in B_x\) with \(#(X) = k + 1\). If \(#(X) = l(x)\), only for \([x]\) is \(g[x] = X\). Otherwise let \(f\) be any \(\leq\)-maximal element of \(X\) and \(Y = X - \{j\}\). Then \(f(Y)\) is unique by induction. For some \(\mu, \nu \in \Sigma^*\),

\[
f(X) = f(Y) \circ (\mu + r_\sigma \circ \nu) \quad \text{and} \quad g(f(X)) = X.
\]

Assume there are also \(\rho, \sigma \in \Sigma^*\) such that \(g(f(Y) \circ (\rho + r_\sigma \circ \sigma)) = X\). Since both \(f(Y) \circ (\mu + r_\sigma \circ \nu)\) and \(f(Y) \circ (\rho + r_\sigma \circ \sigma)\) divide \([x]\), there \([u], [v]\) such that

\[
f(Y) \circ (\mu + r_\sigma \circ \nu) \circ [u] = [x] = f(Y) \circ (\rho + r_\sigma \circ \sigma) \circ [v].
\]

\(f(Y)\) is epic so \((\mu + r_\sigma \circ \nu) \circ [u] = (\rho + r_\sigma \circ \sigma) \circ [v]\). Then there is a unique interchange \(T: (\mu + r_\sigma \circ \nu) \circ u \to (\rho + r_\sigma \circ \sigma) \circ v\).

If the term \((\mu + r_\sigma \circ \nu)\) is never moved by \(T\) then \(\mu = \rho, \nu = \sigma\) and injectivity of \(g\) is established. More precisely, let \(E_T\) be the interchange equivalence class of \(T\). If there is a sequence \((T_1(i_1), \ldots, T_t(i_t))\) in \(E_T\) such that for all \(s \leq t\)

\[
(T_1(i_1) \cdots T_t(i_t)): (\mu + r_\sigma \circ \nu) \circ u \to (\mu + r_\sigma \circ \nu) \circ v
\]

then \(\mu = \rho, \nu = \sigma\). Consider any sequence \(H = (T_1(i_1), \ldots, T_s(i_s))\) in \(E_T\). If \(H\) is null or has the above property we are done. Otherwise, consider the permutations \(p_s\) induced by each initial segment of \(H, (T_1(i_1), \ldots, T_s(i_s))\), \(1 \leq s \leq t\). Consider any maximal position, \(m\), to which \(1\) is moved by the \(p_s\), i.e., any maximal position to which the term \((\mu + r_\sigma \circ \nu)\) is moved during the interchange process determined by \(H\). If \(r_\sigma \circ \nu\) enters position \(m\) by \(L(m)\) it must leave by \(R(m - 1)\) for leaving by \(L(m - 1)\) contradicts uniform representability. Then in \(H\) with \(T_v(i_m)\) the \(L(m - 1)\) and \(T_v(i_v)\) the \(R(m - 1)\)
in question, for \( u < j < v \) no \( T_j(i_j) \) is \( L(m) \) or \( R(m) \) since \( m \) is maximal. If some \( T_j(i_j) = L(m-2) \), choose that with largest index \( j < v \). Then by Lemma 29 (i) there is a sequence \( H' = (T_1(i_1), \ldots, T_u(i_u), \ldots, T_j(i_j), T_v(i_v), T_{j+1}(i_{j+1}), \ldots, T_t(i_t)) \) in \( \mathcal{E}_T \). By Lemma 29 (ii) there is a sequence

\[
H' = (T_1(i_1), \ldots, T_u(i_u), \ldots, T_{j-1}(i_{j-1}), R(m-1), R(m-2), L(m-1), L(m-2), T_{j+1}(i_{j+1}), \ldots, T_t(i_t)).
\]

Continuing for the next \( L(m-2) \) between \( u \) and \( j \), if any, one eventually obtains a sequence in \( \mathcal{E}_T \) in which \( r_j^2 \) does not enter position \( m \) at the \( u \)th step. A similar argument holds for intervening \( R(m-2) \). Then by induction there is a sequence in \( \mathcal{E}_T \) in which \( (\mu + r_j^2 + v) \) is not moved. \( \square \)

The above completes the demonstration that \( g: \mathcal{L}_{[x]} \to B_n \) is an order-isomorphism when \((\Sigma, P)\) is uniformly representable. This completes the proof.

**Theorem 56.** If \((\Sigma, P)\) is uniformly representable then for all \([x]\) is \( S, \mathcal{L}_{[x]}\) is a distributive lattice. Dually, if \((\Sigma, P)\) is uniformly corepresentable then the collection of right divisors of \([x]\) is a distributive lattice. \( \square \)

**References**


