The calculus of strings

D. Terence Langendoen

University of Arizona

War On String May Be Unwinnable, Says Cat General
Headline in *The Onion* 2005-07-27, http://www.theonion.com/content/node/37503

Abstract goes here.

1. String and sequence implication structures

This paper formalizes and applies the notion of the calculus, or logic, of strings described in Ferré 2007: 112.

> The string datatype can be seen as a logic, where formulas are sets of strings …, the deduction relation … is based on … string containment …, and disjunction … computes the maximal substrings shared by 2 strings.

This formalization uses Koslow's (1992) notion of an implication structure I = <S, ⊨>, in which S is a set and ⊨ is an implication relation (Ferré's deduction relation) over S. When S is a set of strings, i.e. a formal language, and ⊨ is the substring relation (Ferré's string containment), I may be called a string implication structure (SIS) with the property that for all s, t ∈ S, s ⊨ t if and only if t is a substring of s (equivalently, s is a superstring of t). More generally, ⊨ satisfies the condition (1).

1. For all $s_1, \ldots s_n, t \in S$: $s_1, \ldots s_n \vDash t$ if and only if t is a substring of a minimal superstring r over $s_1, \ldots s_n$.[1]

The various logical operators are defined for an SIS in the manner of Koslow 1992, as follows. The disjunction, or product, s ∨ t of s, t ∈ S is the least string u ∈ S such that for all v ∈ S, if s ⊨ v and t ⊨ v, then u ⊨ v. That is, u is the least upper bound, or maximal substring, of the disjuncts s, t.[2] The conjunction, or sum, s ∧ t of s, t ∈ S is the least string u ∈ S such that u ⊨ s and u ⊨ t. That is, u is the greatest lower bound, or minimal superstring, of the conjuncts s, t.[3] The negation ¬s of s is the implicationally weakest

[1] A minimal superstring r over $s_1, \ldots s_n$ has each of $s_1, \ldots s_n$ as a substring, and any other candidate string has some r as a substring. It is not required that r belong to S or that it be unique.

[2] The singular 'substring' is used here, in contrast to Ferré's use of the plural 'substrings'; that is, as in standard logic, disjunction is construed here as a logical function (or operator) on strings yielding at most a single value, whereas Ferré construes it as a possibly multi-valued relation.

[3] N-ary products and sums (e.g. $s_1 \vee \ldots \vee s_n$ and $s_1 \wedge \ldots \wedge s_n$) are defined similarly. Throughout this paper, the terms 'product' and 'sum' refer to the results (values) of disjunction and conjunction respectively, 'disjunction' and 'conjunction' to the operators themselves, and 'disjunct' and 'conjunct' to the arguments

string t that together with s entails every string in S. The conditional $s \rightarrow t$ of s and t is the implicationally weakest string that together with s entails t.[4] In addition, modal operators of various sorts are definable for an SIS.

A set $T \subseteq S$ is a sublanguage of S if and only if whenever $s_1, \ldots s_n \in T$ and $s_1, \ldots s_n \vDash t$, $t \in T$. On the other hand, if $s_1, \ldots s_n \vDash t$ and some $s_i \notin T$ $(1 \leq i \leq n)$, then t may or not be a member of T. That is, $\vDash$ preserves sublanguage in the way that ordinary entailment preserves truth in propositional logic. S is, by definition, a sublanguage of itself. The finite sublanguages of S (in addition to S, if S is finite) include, for all $s \in S$, the sets T_s of all substrings of s. If $T_s = \{s, \varepsilon\}$ if $\varepsilon \in S$ (where ε is the empty string) and $T_s = \{s\}$ otherwise, then s is an atomic string in S, and T_s is an atomic sublanguage. Figure 1 shows a relationship between overlapping sublanguages T_u and T_v in a SIS $\langle S, \vDash\rangle$, where the arcs read upwards indicate the entailment relation.

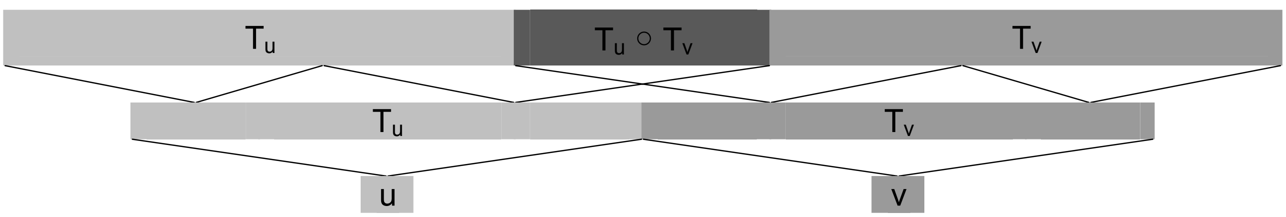

Figure 1. Sublanguages T_u (light gray) and T_v (medium gray) that overlap (dark gray)

$X_s = T_s \cup U_s$ is a chain sublanguage based on a sublanguage T_s, where $U_s = \{s = s_0, s_1, \ldots, s_{i-1}, s_i, \ldots\} \subseteq S$ whose members jointly satisfy the conditions in (2). These conditions insure that s_i is the least upper bound of the pair s_{i-1}, s_i, i.e. that $s_i = s_{i-1} \wedge s_i$, and that there is no other $t \in S$ such that $s_i = s_{i-1} \wedge t$. Figure 2, in which the arcs read leftward indicate entailment in $I = \langle S, \vDash\rangle$, shows a hypothetical chain sublanguage $X_s = T_s \cup U_s$ in which $T_s = \{s, t, u\}$.

2. For all $s_i \in U_s$ $(i > 0)$:
   a. $s_i = s_{i-1} \rightarrow s_i$
   b. $s_i \vDash s_{i-1}$
   c. For all $t \in S$, if t satisfies (2.a) and (2.b), then $t \vDash s_i$.

If S is infinite, then it contains at least one infinite chain sublanguage, unless only finitely many members of S bear the substring relation to one another. If s is atomic in S, then X_s is an atomic chain sublanguage.

---

of the respective operators. The terms 'product' and 'sum' for the results of disjunction and conjunction are taken from the calculus of individuals of Leonard and Goodman (1938), which the calculus of strings greatly resembles.

[4] These operator definitions are all subject to the proviso 'if it exists'; that is the operators may be partial functions on S or not be defined on it at all.

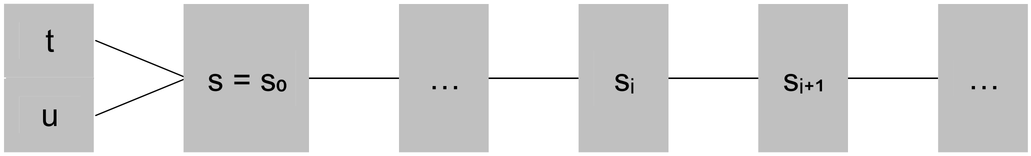

Figure 2. Hypothetical chain sublanguage X_s

A weaker type of implication structure for a set of strings is a sequence implication structure (QIS) I = <S, $\models_Q$>, where S is a set of strings and $\models_Q$ is the subsequence (or interruptible substring) relation with the property that for all s, t $\in$ S, s $\models_Q$ t if and only if t is subsequence of s, either continuous (i.e. a substring) or discontinuous; that is, either s $\models_Q$ t, or $t = r_1 \ldots r_m$ ($m > 1$) and $s = q_0 r_1 q_1 \ldots r_m q_m$ such that $q_1, \ldots q_{m-1}$ are non-null and $q_0 \ldots q_m \in S$.[5] More generally, $\models_Q$ satisfies the condition in (3).

3. For all $s_1, \ldots s_n, t \in S_1$: $s_1, \ldots s_n \models_Q t$ if and only if t is a subsequence of a minimal superstring r over $s_1, \ldots s_n$.

The various logical operators and the notion of sublanguage are defined for a QIS in the same manner as for an SIS.

2. The calculus of regular languages

This section considers SISs and to a lesser extent QISs in which S is a regular language, beginning with infinite regular languages.

2.1. The calculus of infinite regular languages

In I_1 = <S_1, $\models$>, S_1 is the regular language $a^*b^* = \{a^m b^n: m, n \geq 0\} = \{\varepsilon, a, b, aa = a^2, ab, bb = b^2, a^3, a^2b, ab^2, b^3, a^4, a^3b, a^2b^2, ab^3, b^4, \ldots\}$.[6] In I_1, $a^j b^k \models a^p b^q$ if and only if $p \leq j$ and $q \leq k$, so that $a^2b \models ab$ but $a^2b \not\models ab^2$.[7] Since the minimal superstring r over any pair of strings $a^g b^h$, $a^j b^k$ is $a^{\max(g, j)} b^{\max(h, k)}$, $a^g b^h, a^j b^k \models a^p b^q$ if and only if $p \leq \max(g, j)$ and $q \leq \max(h, k)$, so that $a^2b, ab^2 \models a^2b^2$, but $a^2b, ab^2 \not\models ab^3$.

Disjunction and conjunction are total functions on S_1. The product of any pair of disjuncts $a^g b^h$, $a^j b^k$ is $a^g b^h \vee a^j b^k = a^{\min(g, j)} b^{\min(h, k)}$, and the sum of any such pair of conjuncts is $a^g b^h \wedge a^j b^k$ $a^{\max(g, j)} b^{\max(h, k)}$. Three types of products and sums may be distinguished. First, if $g, k > 0$ and $h = j = 0$, or if $h, j > 0$ and $g = k = 0$ (i.e. if one is a

---

[5] It is not required that the individual strings r_i q_j belong to S, but only that their respective concatenations do; cf. Langendoen 2002 for discussion of the 'strict subsequence' relation, which does require the individual strings to belong to S.

[6] I use the more prolix set notation for regular expressions throughout this paper for consistency with set notation not involving regular expressions. ε represents the empty string.

[7] Every entailment in I_{1Q} = <S_1, $\models_Q$> is also valid in I_1 = <S_1, $\models$>, i.e. I_1 and I_{1Q} are equivalent structures.

member of X_{1a} and the other of X_{1b}) the disjuncts and conjuncts (for the remainder of this paragraph, juncts) are disjoint. Their product is ε, and their sum may be called a disjoint sum. For example, $a^2b^2 = a^2 \wedge b^2$ is a disjoint sum. Second, if g = j or h = k, one of the juncts is contained in the other, so that the product or sum is identical to one of its juncts, and the product may be called a contained product, and the sum a contained sum. For example, $a^2b = a^2b \vee a^2b^2$ is a contained product; and $a^2b^2 = a^2b \wedge a^2b^2$ is a contained sum. Otherwise, the juncts partly overlap, and the product may be called an overlapping product, and the sum an overlapping sum. For example, $ab = a^2b \vee ab^2$ is an overlapping product; and in $a^2b^2 = a^2b \wedge ab^2$ is an overlapping sum.

The conditional, likewise, is a total function on S_1. The conditional of a^gb^h as antecedent and a^jb^k as consequent is a^pb^q, where p = j if j > g and p = 0 otherwise, and q = k if k > h and q = 0 otherwise. For example, $a^2 \rightarrow ab = b$, $ab \rightarrow a^2 = a^2$ and $ab \rightarrow a = \varepsilon$. On the other hand, negation is undefined in S_1 since for any string $s \in S_1$, there is no string $t \in S_1$, such that $s, t \vDash u$ for all $u \in S_1$.[8]

Modal operators can also be defined for I_1, such as the box (necessity) modal $\Box a^mb^n = a^mb^n \wedge a^nb^m = a^{\max(m,\, n)}b^{\max(m,\, n)}$ and its counterpart diamond (possibility) modal $\Diamond a^mb^n = a^mb^n \vee a^nb^m = a^{\min(m,\, n)}b^{\min(m,\, n)}$.[9] For example $\Box ab^2 = a^2b^2$, $\Diamond ab^2 = ab$, $\Diamond\Box ab^2 = a^2b^2$ and $\Box\Diamond ab^2 = ab$, and in general $\Box s \vDash s \vDash \Diamond s$ and $\Diamond\Box s \vDash \Box\Diamond s$ for all $s \in S_1$.

Figure 3 diagrams the top part of I_1; its arcs, when understood as pointing upward, show all the one-premise non-reflexive entailments among the strings of S_1 of length < 4 and some for those of length 4.[10]

---

[8] This observation about negation holds for any SIS in which S is infinite, but not for its dual; see note 10.

[9] $\Box$ is a necessity modal in I_1, since for all $s, t \in S_1$, $\Box(s \wedge t) \Leftrightarrow \Box s \wedge \Box t$ and $\Box s \vee \Box t \vDash \Box(s \vee t)$, but there are $s, t \in S_1$ such that $\Box(s \vee t) \nvDash \Box s \vee \Box t$, e.g. a^2b, ab^2, since $\Box(ab^2 \vee a^2b) = \Box ab = ab$, whereas $\Box ab^2 \vee \Box a^2b = a^2b^2 \vee a^2b^2 = a^2b^2$, and $ab \nvDash a^2b^2$. $\Diamond$ is a possibility modal in I_1, since for all $s, t \in S_1$, $\Diamond s \vee \Diamond t \Leftrightarrow \Diamond(s \vee t)$ and $\Diamond(s \wedge t) \vDash \Diamond s \wedge \Diamond t$, but there are $s, t \in S_1$ such that $\Diamond s \wedge \Diamond t \nvDash \Diamond(s \wedge t)$, e.g. a^2b, ab^2, since $\Diamond ab^2 \wedge \Diamond a^2b = ab \wedge ab = ab$, whereas $\Diamond(ab^2 \wedge a^2b) = \Diamond a^2b^2 = a^2b^2$, and $ab \nvDash a^2b^2$. However $\Box$ and $\Diamond$ are not interdefinable using negation in the usual way since negation is undefined in I_1. Both $\Box$ and $\Diamond$ map S_1 onto the context-free language $S_5 = \{a^nb^n: n \geq 0\} \subset S_1$ discussed below in section 3.

[10] Reading the arcs downward, Figure 3 represents the bottom part of the dual SIS $I_{1^\wedge} = \langle S_1, \vDash^\wedge \rangle$ in which for all $s_1, \ldots s_n, t \in S_1$, $s_1, \ldots s_n \vDash^\wedge t$ if and only if t is a superstring of a maximal substring q over $s_1, \ldots s_n$. Conjunction in $I_{1^\wedge}$ is equivalent to disjunction in I_1, and vice versa. Also in $I_{1^\wedge}$ negation is a partial function on S_1: $\neg a^mb^n = \varepsilon$ if m, n > 0; otherwise $\neg a^mb^n$ is undefined.

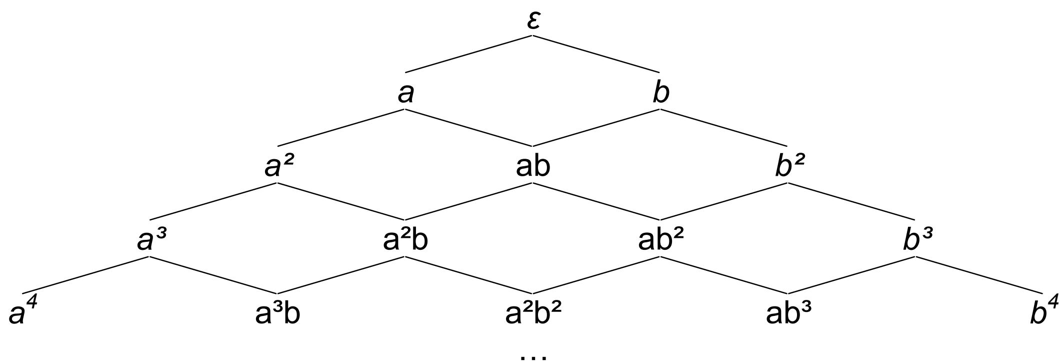

Figure 3. I_1 for the regular language $S_1 = \{a^m b^n: m, n \geq 0\}$

The infinite sublanguages of S_1, in addition to S_1 itself, are all of the form $\{a^m b^n: m \geq 0, 0 \leq n \leq q \text{ or } n \geq 0, 0 \leq m \leq p\}$, which may be more perspicuously represented as $a^*b^{\leq q} \mid a^{\leq p}b^*$. By setting p = q = 0 for each string in a sublanguage, the atomic chain sublanguages based on a and b are obtained, namely $X_{1a} = \{a^m: m \geq 0\}$ and $X_{1b} = \{b^n: n \geq 0\}$, which are proper subsets of every other infinite sublanguage of S_1. Their intersection is the singleton {ε} and their union $S^*_1 = X_{1a} \cup X_{1b}$ is a proper subset of S_1.

S_1 is closed under conjunction in S^*_1; i.e. every $s \in S_1$ is the sum of a pair $t, u \in S^*_1$,[11] and for every $s \notin S_1$, there is no pair $t, u \in S^*_1$ such that s is their sum.[12] The members of S^*_1, italicized in Figure 3, are the conjunctive generators of S_1, and each member of the complement $S^{*\prime}_1 = S_1 - S^*_1 = \{a^m b^n: m, n > 0\}$ is the disjoint sum of a single pair of generators only.[13] Consequently, S_1 is structurally unambiguous in I_1: Every member of S_1 is either a generator or the disjoint sum of a single pair of generators. In addition, S_1

[11] If t, u are both drawn from X_{1a} or from X_{1b}, then s = t or s = u; for example, if t = b and u = b^2, then $s = b \wedge b^2 = b^2 = t$. Otherwise if $t \in X_{1a}$ and $u \in X_{1b}$, then s = tu, and vice versa; for example, if $t = a^2$ and $u = b^3$, then $s = a^2b^3 = tu$.

[12] Only strings over {a, b} that are not in S_1, such as ba, need be considered. If $ba = a \wedge b$, then $ab \neq a \wedge b$, since conjunction is a function. This is a contradiction, since $ab = a \wedge b$ in S_1. Therefore $ba \neq a \wedge b$.

[13] Since the only generators considered in this paper are conjunctive ones, the term 'generator' is used henceforth to refer to a conjunctive generator only.

is closed under conjunction in S_1 as a whole, but every member of $S^{*\prime}_1$ except for ab is an overlapping sum of at least one pair of members of S_1.[14]

Every language like S^*_1 in $I^*_1 = \langle S^*_1, \vDash\rangle$ in Figure 4 that consists entirely of members of its atomic chain sublanguages is identical to its generator set.[15]

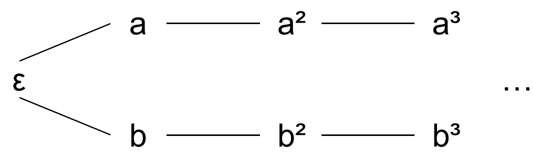

Figure 4. I^*_1 for $S^*_1 = \{a^m \mid b^n: m, n \geq 0\}$

Next, $I_2 = \langle S_2, \vDash\rangle$ in Figure 5 contains the regular language $S_2 = \{a^m b^n c^p: m, n, p \geq 0\} = \{\varepsilon, a, b, c, a^2, ab, ac, b^2, bc, c^2, a^3, a^2b, a^2c, ab^2, ab^2, abc, ac^2, b^3, b^2c, bc^2, c^3, ...\}$ whose generator set is $S^*_2 = \{a^m \mid b^n \mid c^p: m, n, p \geq 0\}$, italicized in Figure 5. Unlike S_1, S_2 is structurally ambiguous, since every member of the complement $S^{*\prime}_2$ of the generator set of the form $a^m b^n c^p$ ($m, n, p > 0$) can be expressed as a disjoint sum in three different ways; e.g. $abc = a \wedge (b \wedge c) = a \wedge bc$; $abc = (a \wedge b) \wedge c = ab \wedge c$; and $abc = a \wedge b \wedge c$, corresponding to the structural ambiguity of three-conjunct coordination in English in which phrases of the form A and B and C can be bracketed [A and [B and C]], [[A and B] and C] and [A and B and C]. Expressing abc as the overlapping sum of ab, bc neutralizes the structural ambiguity.

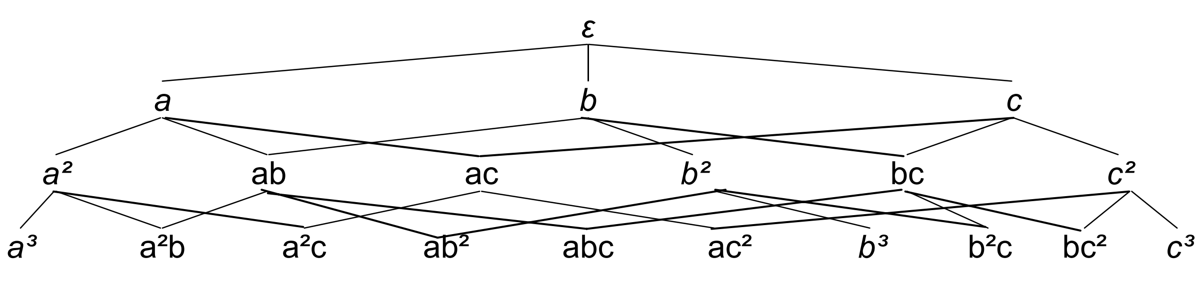

Figure 5. I_2 for $S_2 = \{a^m b^n c^p: m, n, p \geq 0\}$

Finally, $I_3 = \langle S_3, \vDash\rangle$ in Figure 6, in which $S_3 = \{(a \mid b)^n: n \geq 0\} = \{\varepsilon, a, b, a^2, ab, ba, b^2, a^3, a^2b, aba, ab^2, ba^2, bab, b^2a, b^3, \ldots\}$, is the universal language over the vocabulary {a, b}.

---

[14] For example, $a^2b^3 = a^2b^2 \wedge ab^3 = a^2b \wedge ab^3 = a^2 \wedge ab^3 = a^2b^2 \wedge b^3 = a^2b \wedge b^3$. The pair a^2b^2, ab^3 are the maximal overlapping conjuncts of a^2b^3, as one of them must be a conjunct of every overlapping conjunction of which a^2b^3 is the sum.

[15] The arcs in Figure 4 are understood to point to the left, just as in Figure 2. Conjunction is a partial function in I^*_1, since every pair x, y in which $x \in \{a^m: m > 0\}$ and $y \in \{b^n: n > 0\}$ lacks a greatest lower bound.

The atomic chain sublanguages of S_3 are the same as for S_1, namely $X_{3a} = \{a^m: m \geq 0\}$ and $X_{3b} = \{b^n: n \geq 0\}$, so all the members of those sublanguages belong to the generator set of S_3. Moreover, because of the non-commutativity of concatenation, no member of the complement of $X_{3a} \cup X_{3b}$, namely $S^{*\prime}_{3ab} = \{x \in S_3: x \vDash ab \text{ or } x \vDash ba\}$, is a discrete or overlapping sum in S_3, and no pair of logically independent members of $S^{*\prime}_{3ab}$ has a product in S_3. For example, whereas $aba = ab \wedge aba$ is a contained sum in S_3, and $ab = ab \vee aba$ is a contained product, the pair a, b has no sum because both ab and ba are candidate greatest lower bounds, but neither is a substring of the other, and the pair ab, ba has no product, because both a and b are candidate least upper bounds, but neither is a superstring of the other.[16]. Because of the failure of conjunction in $S^{*\prime}_{3ab}$, every member of that set also belongs to the generator set of S_3, from which it follows that S_3 is co-extensive with its generator set. There are analogs to I_3 that are closed under disjunction and conjunction and for which the generator set is a proper subset of the set as a whole, but the languages of such SISs are context free; see section 3.2 for discussion of such an analog.

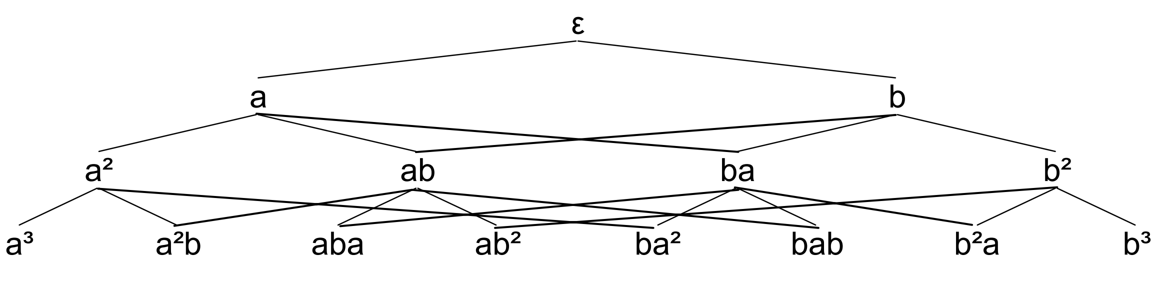

Figure 6. I_3 for $S_3 = \{(a \mid b)^n: n \geq 0\}$, showing the results of the partial failure of conjunction

2.2. The calculus of finite languages

I_1, I^*_1, I_2, and I_3 are SISs over infinite regular languages. $I_{1f} = \langle S_{1f}, \vDash\rangle$, in which $S_{1f} = \{a^m b^n: m, n \geq 0, m+n \leq 4\}$, is a finite SIS, represented in its entirety by Figure 3 omitting the ellipsis at the bottom. Disjunction and the conditional are total functions on S_{1f}, but conjunction and negation are partial ones. Conjunction is defined for every pair which has a greatest lower bound in I_{1f} such as a^2, ab^2, since $a^2 \wedge ab^2 = a^2b^2 \in S_{1f}$, but undefined for all others, such as a^3, ab^2, which have no greatest lower bound in S_{1f}. Negation is defined for a^4 and b^4 (they are each other's negations), but is undefined for every other $s \in S_{1f}$. The generator set of S_{1f} is $S^*_{1f} = \{a^m \mid b^n: m, n \leq 4\}$, and its

---

[16] However according to Ferré, both a and b would be least upper bounds for the pair ab, ba in S_3, so that disjunction would not be a function at all in S_3, but simply a relation.

complement $S^{*\prime}_{1f} = \{a^m b^n: m, n > 0, m+n \leq 4\}$. On the assumption that, for example, a^3b^2 is the sum of a^3, ab^2, S_{1f} is not closed under conjunction of members of S^*_{1f}, since a^3, $ab^2 \in S_{1f}$, but $a^3b^2 \notin S_{1f}$.[17]

The finite SIS $I_{1ab} = \langle S_{1ab}, \models\rangle$ in Figure 7, in which $S_{1ab} = \{\varepsilon, a, b, ab\}$, the sublanguage of S_1 for the string ab, is the only classical (boolean) sublanguage SIS of I_1 other than I_ε, in which the laws of double negation and excluded middle both hold. All the other SISs for sublanguages of S_1 are nonclassical, for example $I_{1a^2b^2} = \langle S_{1a^2b^2}, \models\rangle$ in Figure 8, in which $S_{1a^2b^2} = \{\varepsilon, a, b, a^2, ab, b^2, a^3, a^2b, ab^2, a^2b, ab^2, a^2b^2\}$, the sublanguage of S_1 for the string a^2b^2. Disjunction, conjunction, negation and the conditional are all total functions on $S_{1a^2b^2}$, but the laws of double negation and excluded middle both fail in $I_{1a^2b^2}$. Double negation fails because (for example) $\neg\neg a^2b = \neg b^2 = a^2$, not a^2b. Excluded middle fails because $a^2b \vee \neg a^2b = a^2b \vee b^2 = b$, not ε. Like all sublanguages of a language that is closed under conjunction, both S_{1ab} and $S_{1a^2b^2}$ are closed under conjunction.

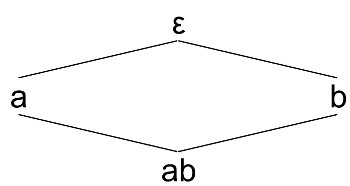

Figure 7. I_{1ab} for S_{1ab}, the sublanguage of S_1 for the string ab

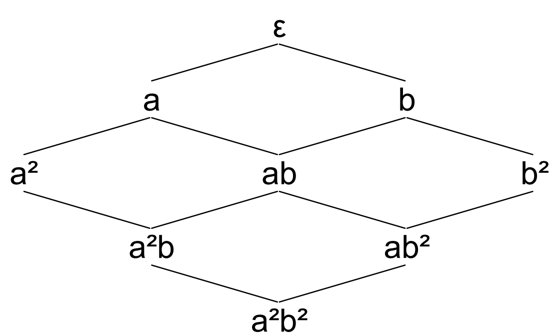

Figure 8. $I_{1a^2b^2}$ for $S_{1a^2b^2}$, the sublanguage of S_1 for the string a^2b^2

Finally, we consider a series of finite SISs and QISs that illustrate a variety of conditions under which structural ambiguity does or does not arise in such structures. As noted above, in the infinite regular language SIS I_2, certain strings are three-ways ambiguous,

[17] The assumption, however, depends on conjunction having the same properties outside of S_{1f} as within it. Taking into consideration all the strings that do not belong to S_{1f}, the pair a^3, ab^2 has no conjunction, since a^3b^2 and aba^3 are candidates, and neither is a substring of the other. So it can be argued that S_{1f} is closed under conjunction because of this technicality.

being disjoint sums in three different ways. The finite language SIS $I_{4a} = \langle S_{4a}, \vDash\rangle$ and QIS $I_{4aQ} = \langle S_{4a}, \vDash_Q\rangle$, in which S_{4a} = {a, b, c, ab, ac, bc, abc}, the set of all substrings of abc except ε, are shown together in Figure 9, in which solid arcs indicate entailments in both structures, and dashed arcs entailments in the QIS only, a convention followed throughout this paper whenever an SIS and a QIS are diagrammed together.[18] The generator set of both structures is S^*_{4a} = {a, b, c}. In I_{4a}, the string abc manifests the three-way ambiguity of I_2, since abc = a ∧ bc = ab ∧ c = a ∧ b ∧ c, and all other members of S_{4a} are unambiguous. In I_{4aQ}, the string abc is four-ways ambiguous, since abc = b ∧ ac in I_{4aQ} as well. A similar result holds for I_{2Q}; every string that is three-ways ambiguous in I_2 is four-ways ambiguous in $I_{2Q} = \langle S_2, \vDash_Q\rangle$.

By removing the string b from S_{4a}, resulting in S_{4b} = {a, c, ab, ac, bc, abc}, the SIS I_{4b} and QIS I_{4bQ} in Figure 10 are obtained, with the generator set S^*_{4b} = {a, c, ab, bc}. In both structures, abc = a ∧ bc = ab ∧ c and so is two-ways ambiguous. The further removal of the string ac has no effect on the ambiguity of abc in the resulting SIS and QIS, with the latter collapsing onto the former, as in Figure 11 for the SIS $I_{4c} = \langle S_{4c}, \vDash\rangle$ in which S_{4c} = {a, c, ab, bc, abc} and in which abc = a ∧ bc = ab ∧ c as before.

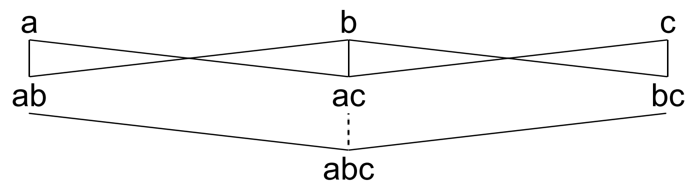

Figure 9. I_{4a} and I_{4aQ} for the ambiguous language S_{4a}

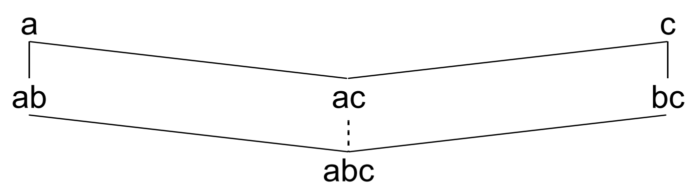

Figure 10. I_{4b} and I_{4bQ} for the ambiguous language S_{4b}

[18] Adding ε to S_{4a} yields S_{2abc}, the sublanguage of S_2 for the string abc. The QIS $I_{2abcQ} = \langle S_{2abc}, \vDash_Q\rangle$ is classical, but the SIS $I_{2abc} = \langle S_{2abc}, \vDash\rangle$ is not. Double negation fails because ¬ac = abc and ¬abc = ε, so that ¬¬ac = ε, not ac. Excluded middle fails because the disjunction of ac and ¬ac (= abc) is undefined. Both a and c are candidates, but neither is a superstring of the other.

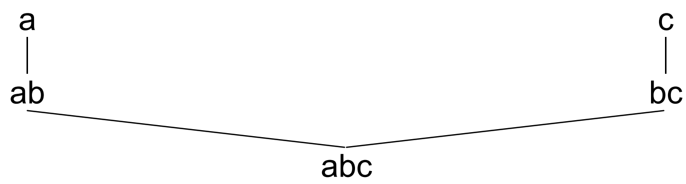

Figure 11. I_{4c} for the ambiguous language S_{4c}

If c is replaced by b, even if the string ac is included, the resulting SIS $I_{4d} = \langle S_{4d}, \models\rangle$ and QIS $I_{4dQ} = \langle S_{4d}, \models_Q\rangle$ in Figure 12, in which S_{4d} = {a, b, ab, ac, bc, abc} and whose generator set is $S^*_{4d} = S_{4d}$ - {abc}, are unambiguous. In both I_{4d} and I_{4dQ}, abc = a $\wedge$ bc only as a disjoint sum. However if a is replaced by b in S_{4c} and the string ac is included, the resulting SIS $I_{4e} = \langle S_{4e}, \models\rangle$ in Figure 13, in which S_{4e} = {b, c, ab, ac, bc, abc} and whose generator set is S^*_{4e} = {b, c, ab, ac}, is unambiguous, but the resulting QIS $I_{4eQ} = \langle S_{4e}, \models_Q\rangle$ is two-ways ambiguous. In I_{4e}, abc = ab $\wedge$ c only as a disjoint sum, whereas in I_{4eQ}, abc = ab $\wedge$ c = ac $\wedge$ b. There is no 'right' answer to the question "Is the string abc structurally ambiguous in the language S_{4e}?" It depends on the implication structure it occurs in. In I_{4e}, it is unambiguous, but in I_{4eQ}, it is ambiguous. There would be a right answer for a counterpart to S_{4e} occurring as a sublanguage of a natural language, if there were empirical evidence concerning the ambiguity of the counterpart to abc.

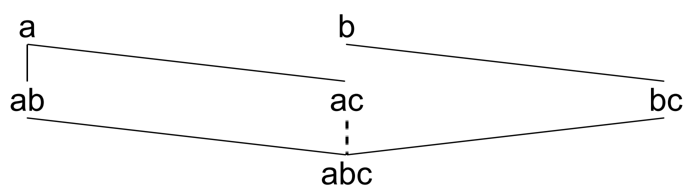

Figure 12. I_{4d} and I_{4dQ} for the unambiguous language S_{4d}

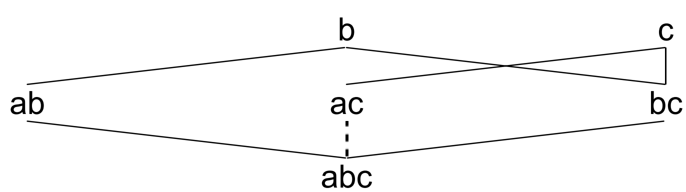

Figure 13. I_{4e} and I_{4eQ} for S_{4e}, which is unambiguous in I_{4e} but ambiguous in I_{4eQ}

Further, if the string b is removed from S_{4d}, yielding S_{4f} = {a, ab, ac, bc, abc}, and from S_{4e}, yielding S_{4g} = {c, ab, ac, bc, abc}, the resulting SISs and QISs are unambiguous. In both I_{4f} and I_{4fQ}, abc = a $\wedge$ bc only as a disjoint sum, and in both I_{4g} and I_{4gQ}, abc = ab $\wedge$ c only. However, if the string a is removed from S_{4d} yielding S_{4h} = {b, ab, ac, bc, abc},

abc = b ∧ ac as a disjoint sum in I_{4hQ} in Figure 14, but abc is not a disjoint sum at all in I_{4h}; i.e. there is no proper bracketing for it.[19]

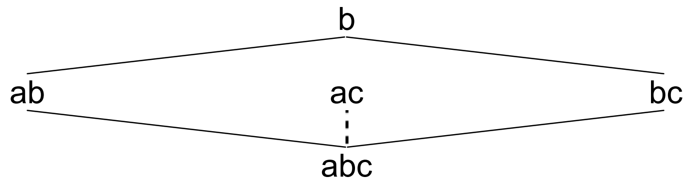

Figure 14. I_{4h} and I_{4hQ} for S_{4h}, in which abc has a unique disjoint sum in I_{4hQ} but not in I_{4h}

Next, starting again with S_{4a} and removing bc yields S_{4i}, and removing ab yields S_{4j}; the structures $I_{4i} = \langle S_{4i}, \models \rangle$ and $I_{4j} = \langle S_{4j}, \models \rangle$ are two-ways structurally ambiguous, whereas the structures $I_{4iQ} = \langle S_{4i}, \models_Q \rangle$ and $I_{4jQ} = \langle S_{4j}, \models_Q \rangle$ are three-ways structurally ambiguous; In $I_{4i} = \langle S_{4i}, \models \rangle$ and $I_{4jQ} = \langle S_{4j}, \models_Q \rangle$, in Figure 15 abc = ab ∧ c = a ∧ b ∧ c; in addition in I_{4jQ}, abc = ac ∧ b. On the other hand, removing ac from S_{4a} yields S_{4k}, which is three-ways structurally ambiguous in both $I_{4k} = \langle S_{4k}, \models \rangle$ and $I_{4kQ} = \langle S_{4k}, \models_Q \rangle$ in Figure 16; abc = ab ∧ c = a ∧ bc = a ∧ b ∧ c in both structures.

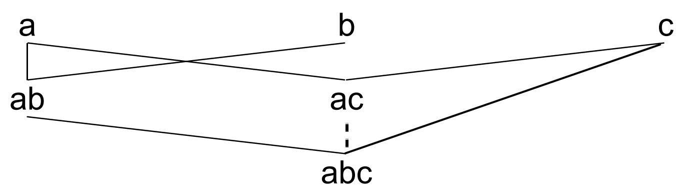

Figure 15. I_{4i} and I_{4iQ} for S_{4i}; in I_{4i}, abc is two-ways structurally ambiguous; in I_{4iQ} it is three-ways structurally ambiguous

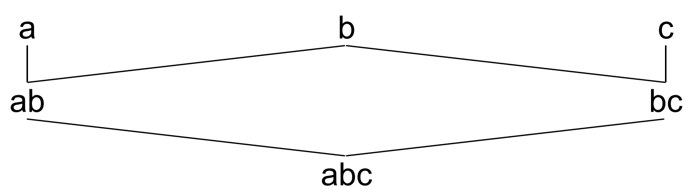

Figure 16. I_{4k} and I_{4kQ} for S_{4i}; in both of which abc is three-ways structurally ambiguous

If ab and bc are removed from S_{4a}, yielding S_{4l}, the structures $I_{4l} = \langle S_{4l}, \models \rangle$ and $I_{4lQ} = \langle S_{4l}, \models_Q \rangle$ in Figure 17 are obtained; in the former, abc is unambiguous, since its only analysis as a disjoint sum is as a ∧ b ∧ c; however abc is two-ways ambiguous in the latter, since there it also has the analysis ac ∧ b. On the other hand, if bc and ac are

[19] The string ac is neither a substring nor a superstring of any other string in I_{4h}, i.e. it is logically independent.

removed, retaining ab, or if ab and ac are removed, retaining bc, the resulting SIS and QIS are equivalent, and are two-ways ambiguous. Finally, if ab, ac and bc are all removed, resulting in S_{4m} = {a, b, c, abc}, the resulting SIS I_{4m} and QIS I_{4mQ} in Figure 18 are again equivalent and are unambiguous; $a \wedge b$, $a \wedge c$, $b \wedge c$, and $a \wedge b \wedge c$ are all equivalent to abc.

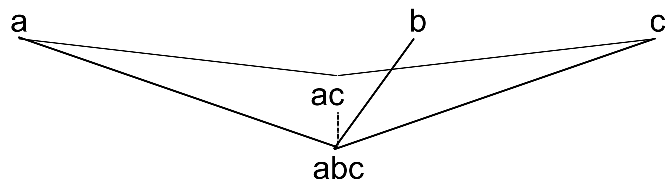

Figure 17. I_{4l} and I_{4lQ} for S_{4l}; in I_{4l}, abc is unambiguous; in I_{4lQ} it is two-ways ambiguous

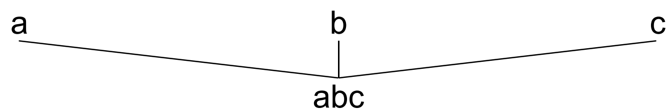

Figure 18. $I_{4m} \equiv I_{4mQ}$ for S_{4m}, in which abc is unambiguous

3. The calculus of context-free languages

This section describes SISs and QISs for context-free languages. First, $I_5 = \langle S_5, \vDash\rangle$ in Figure 19, is the SIS in which S_5 is the context-free language $\{a^n b^n: n \geq 0\}$. Conjunction and disjunction are total functions in I_5, but S_5 is identical to its atomic chain sublanguage X_{5ab}. Consequently, the generator set S^*_5 of S_5 is also identical to it, analogous to the situation in I^*_1 for the regular language $S^*_1 = \{a^m \mid b^n: m, n \geq 0\}$. However, if the equality constraint on the number of a's and b's in S_5 is relaxed, the generator sets become context-free subsets of the sets as a whole and the resulting structures approximate, but never reach, that of I_1 for the regular language $I_1 = \{a^m b^n: m, n \geq 0\}$, as shown in Figure 20 and Figure 21 for the first two steps in the approximation: $I_{5\text{-}1} = \langle S_{5\text{-}1}, \vDash\rangle$, in which $S_{5\text{-}1} = \{a^m b^n: m, n \geq 0, |m\text{-}n| \leq 1\}$, and $I_{5\text{-}2} = \langle S_{5\text{-}2}, \vDash\rangle$, in which $S_{5\text{-}2} = \{a^m b^n: m, n \geq 0, |m\text{-}n| \leq 2\}$. The generator set of $S_{5\text{-}1}$ is $S^*_{5\text{-}1} = \{a^m b^n: m, n \geq 0, |m\text{-}n| = 1\} \cup \{\varepsilon\}$, italicized in Figure 20, and its complement is $S^{*\prime}_{5\text{-}1} = \{a^n b^n: n > 0\}$. The generator set of $S_{5\text{-}2}$ is $S^*_{5\text{-}2} = \{a^m b^n: m, n \geq 0, |m\text{-}n| = 2\} \cup \{\varepsilon, a, b\}$, italicized in Figure 21, and its complement is $S^{*\prime}_{5\text{-}2} = \{a^m b^n: m, n > 0; |m\text{-}n| \leq 1$. Thus the strings containing equal or nearly equal numbers of a's and b's are disjoint sums in the manner of the regular language SIS I_1, e.g. $ab = a \wedge b$ in both $S_{5\text{-}1}$ and $S_{5\text{-}2}$, and $a^2b = a^2 \wedge b$, $ab^2 = a \wedge b^2$ and $a^2b^2 = a^2 \wedge b^2$ in $S_{5\text{-}2}$ alone.

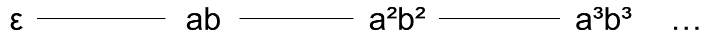

Figure 19. I_5 for the context-free language $S_5 = \{a^n b^n: n \geq 0\}$

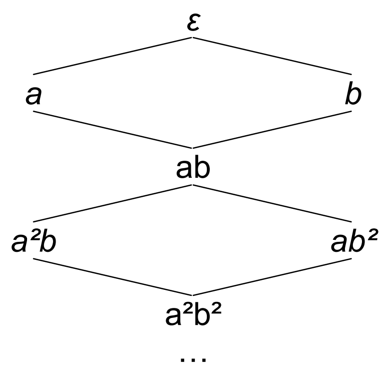

Figure 20. I_{5-1} for $S_{5-1} = \{a^m b^n: m, n \geq 0, n\text{-}1 \leq m \leq n\text{+}1\}$

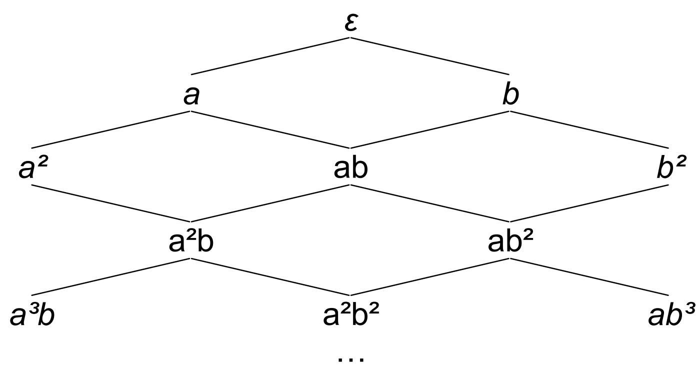

Figure 21. I_{5-2} for $S_{5-2} = \{a^m b^n: m, n \geq 0, n\text{-}2 \leq m \leq n\text{+}2\}$

Next SISs for two context-free mirror-image languages are presented. $I_6 = <S_6, \vDash>$ in Figure 22, is the SIS in which $S_6 = \{xy : x \in \{a^m b^n: m, n \geq 0\}; y \in \{d^n c^m: m, n \geq 0\}$, the mirror image of x with c in place of a and d in place of b$\} = \{\varepsilon, ac, bd, a^2c^2, abdc, b^2d^2, a^3c^3, a^2bdc^2, ab^2d^2c, b^3d^3, \ldots\}$. Conjunction is a partial function in I_6 and the generator set S^*_6 is identical to S_6, since every member of S_6 belongs to some chain sublanguage of S_6, e.g. $a^3c^3 \in X_{6ac}$, $a^2bdc^2 \in X_{6bd1}$, $ab^2d^2c \in X_{6b^2d^2}$, and $b^3d^3 \in X_{6bd2}$, and S^*_6 is the union of those sublanguages.[20] $I_7 = <S_7, \vDash>$ in Figure 23 is the SIS in which $S_7 = \{xy: x \in \{(a \mid b)^n: n \geq 0\}, y \in \{(c \mid d)^n: n \geq 0\}$, the mirror image of x with c in place of a and d in place of b$\} = \{\varepsilon, ac, bd, a^2c^2, bacd, abdc, b^2d^2, a^3c^3, ba^2c^2d, abacdc, b^2acd^2, a^2bdc^2, babdcd, ab^2d^2c, b^3d^3, \ldots\}$. I_7 has a binary tree configuration, so that every string in S_7 is the disjunction of its daughters, e.g. $abdc \vee b^2d^2 = bd$ as in S_6, and $a^2bdc^2 \vee babdcd = abdc$, as well as of any pair of its descendants on different branches that are not co-

[20] S_6 has as many atomic chain sublanguages as there are paths through the tree in Figure 22.

daughters, e.g. $a^2bdc^2 \vee b^2d^2 = bd$ and $a^2bdc^2 \vee ba^2c^2d = \varepsilon$. Conjunction, however, is a partial function in S_7, and the generator set S^*_7 is identical to S_7 itself, just as in the case of S_3 in I_3.

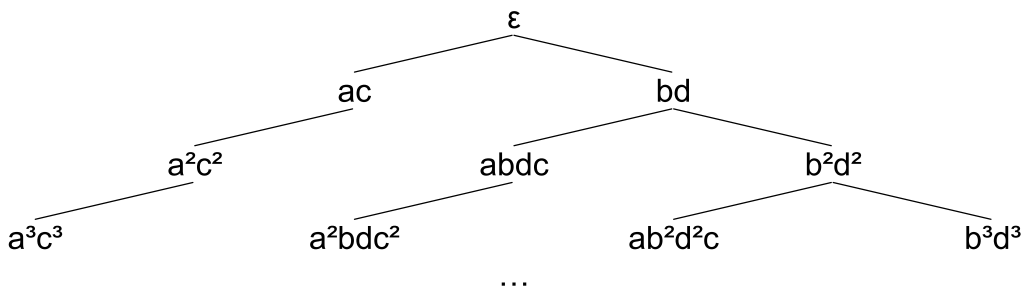

Figure 22. I_6 for $S_6 = \{xy: x \in \{a^mb^n: m, n \geq 0\}; y \in \{d^nc^m: m, n \geq 0\}$, the mirror image of x with c in place of a and d in place of b}

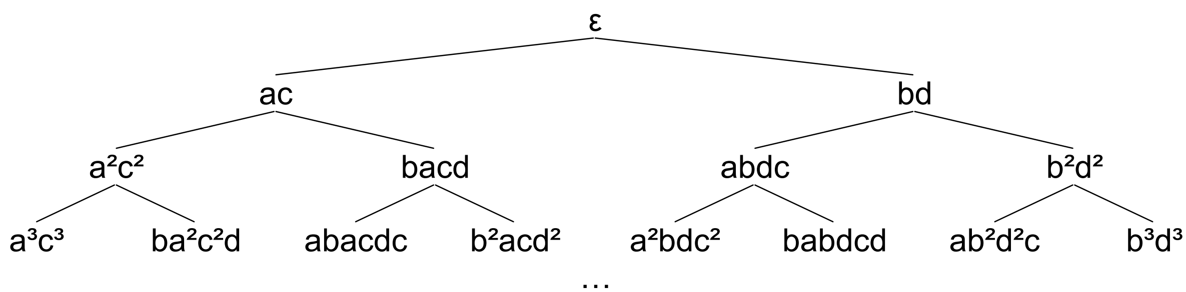

Figure 23. I_7 for $S_7 = \{xy: x \in \{(a \mid b)^n: n \geq 0\}; y \in \{(c \mid d)^n: n \geq 0\}$, the mirror image of x as in S_6}

3.1. Sequence implication structures for context-free languages

However, conjunction is a total function in the QIS $I_{6Q} = \langle S_6, \vDash_Q \rangle$ in Figure 24 and the QIS $I_{7Q} = \langle S_7, \vDash_Q \rangle$ in Figure 25 that correspond to I_6 and I_7 respectively. In I_{6Q}, $a^2bdc^2 \vDash_Q a^2c^2$, since $a^2c^2 = r_1r_2$ where $r_1 = a^2$, $r_2 = c^2$; and $a^2bdc^2 = q_0r_1q_1r_2q_2$ where $q_0 = q_2 = \varepsilon$ and $q_1 = bd$, so that $q_0q_1q_2 = bd \in S_6$. The QIS I_{6Q} is isomorphic to the SIS I_1 for the regular language S_1 from which the context-free language S_6 is obtained by mirroring, and the QIS I_{7Q} is isomorphic to the SIS I_3 for the regular language S_3 from which the context-free language S_7 is obtained by mirroring. Consequently, the generator set for S_6 in I_{6Q} is the context-free language $S^*_{6Q} = \{a^mc^m: m \geq 0\} \cup \{b^nd^n: n \geq 0\}$, italicized in Figure 24, and its complement the context-free language $S^{*\prime}_{6Q} = \{xy: x \in \{a^mb^n: m, n > 0\}; y \in \{d^nc^m: m, n > 0\}$, the mirror image of x as in S_6}. For S_7 in I_{7Q}, however, the generator set is identical to S_7 as a whole, just as it is for S_3 and for the same reason. There are analogs to I_{7Q} that are closed under disjunction and conjunction and for which the generator set is a proper subset of the set as a whole, but the languages of such SISs are context sensitive; see section 4.2 for discussion of such an analog.

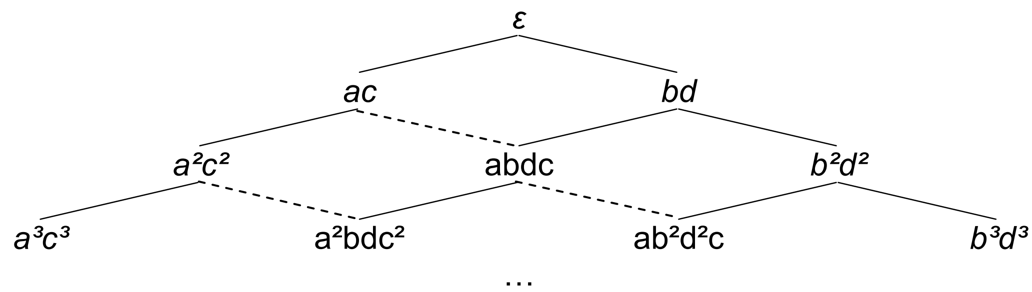

Figure 24. I_{6Q} for S_6 with I_6 superimposed; cf. Figure 3

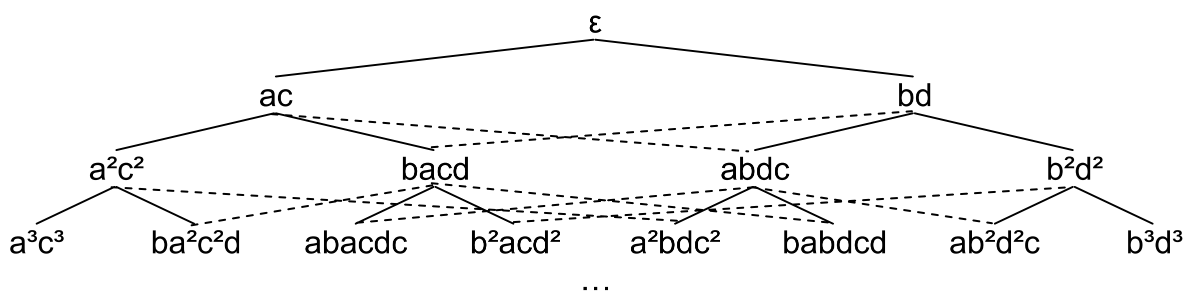

Figure 25. I_{7Q} for S_7 with I_7 superimposed; cf. Figure 6

3.2. The calculus of inherently ambiguous context-free languages

Certain context-free languages are inherently ambiguous (Parikh 1961, Chomsky 1963: 389), in the sense that certain of their members are structurally ambiguous with respect to every context-free grammar that generates them; i.e. each such string must have at least two structural descriptions, or bracketings. For example, in $S_{8n} = \{a^m b^n c^p: m, n, p \geq 0, m = n \text{ or } n = p\} = \{\varepsilon, a, c, a^2, ab, bc, c^2, a^3, abc, c^3, a^4, a^2b^2, a^2bc, abc^2, b^2c^2, c^4, \ldots\}$ in $I_{8n} = \langle S_{8n}, \vDash\rangle$ in Figure 26, every string of the form $a^k b^k c^k$ ($k > 0$) receives two bracketings, $[a^k [b^k c^k]]$ and $[[a^k b^k] c^k]$, with respect to every context-free grammar that generates S_{8n}. The generator set S^*_{8n} for I_{8n} is the regular language $\{a^m: m \geq 0\} \cup \{c^p: p \geq 0\} \cup \{ab, bc\}$, the rest of the language being sums of members of S^*_{8n} or of other sums, some of them not disjoint. For example, $abc^2 = ab \wedge c^2$, $a^2b^2 = a^2 \wedge ab$ and $a^3b^3 = a^3 \wedge a^2b^2$. The structurally ambiguous members of S_{8n}, and only those, are disjoint sums in two different ways that exactly match the bracketings; for example, $abc = a \wedge bc = ab \wedge c$, and $a^2b^2c^2 = a^2 \wedge b^2c^2 = a^2b^2 \wedge c^2$.

However the same correspondence of structural bracketings with disjoint sums does not occur for the inherently ambiguous languages $S_{8m} = \{a^m b^n c^p: m, n, p \geq 0, m = n \text{ or } m = p\}$ and $S_{8p} = \{a^m b^n c^p: m, n, p \geq 0, m = p \text{ or } n = p\}$. Choosing S_{8m} to illustrate, every string of the form $a^k b^k c^k$ ($k > 0$) receives the bracketings $[[a^k b^k] c^k]$ and $[a^k [b^k] c^k]$ with respect to every context-free grammar that generates that language. However, those strings are the disjoint sums in only one way, corresponding to the first of these bracketings only, in the SIS I_{8m} = <S_{8m}, ⊨> in Figure 27; e.g. abc = ab ∧ c, but abc ≠ ac ∧ b because ac is not a substring of abc. However a^kc^k is a subsequence of $a^kb^kc^k$, so in the QIS I_{8mQ} = <S_{8m}, $\vDash_Q$>, abc = ab ∧ c = ac ∧ b as desired.

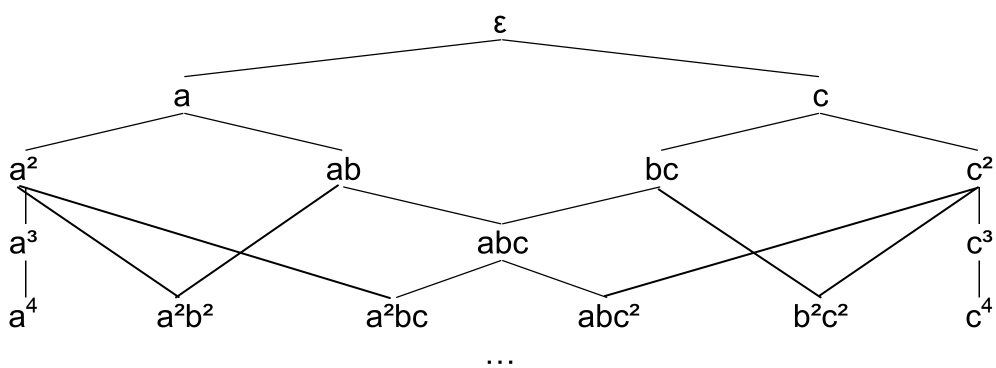

Figure 26. I_{8n} for the inherently ambiguous context-free language S_{8n} = {$a^mb^nc^p$: m, n, p ≥ 0, m = n or n = p}

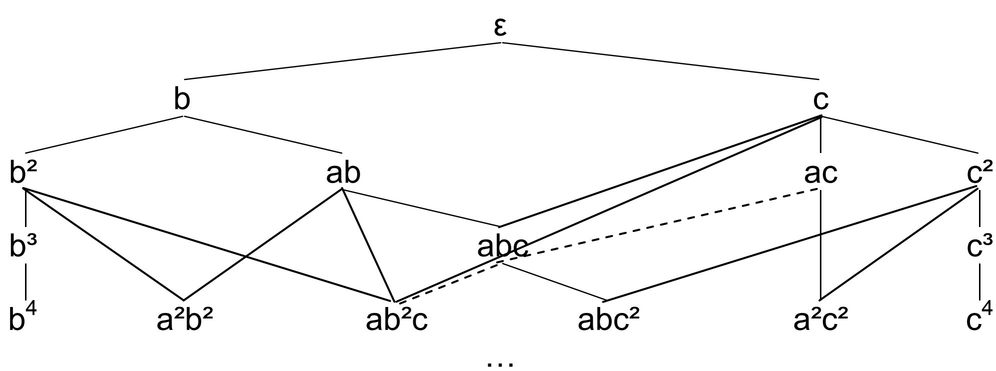

Figure 27. I_{8m} and I_{8mQ} for the inherently ambiguous context-free language S_{8m} = {$a^mb^nc^p$: m, n, p ≥ 0, m = n or m = p}

The situation is different again for the inherently ambiguous context-free language S_9 = {$a^mb^nc^p$: m, n, p ≥ 0, m = n or m = p or n = p} = {ε, a, b, c, a^2, ab, ac, bc, b^2, c^2, a^3, abc, b^3, c^3, …} in the SIS I_9 = <S_9, ⊨> in Figure 28. All strings of the form $a^kb^kc^k$ (k > 0) in S_9 have three bracketings with respect to every context-free grammar that generates S_9, namely [a^k [b^k c^k]], [[a^k b^k] c^k] and [a^k [b^k] c^k]. Each of those strings is also a disjoint sum in three different ways in I_9: $a^kb^kc^k$ = a^k ∧ b^kc^k = a^kb^k ∧ c^k = a^k ∧ b^k ∧ c^k for all k > 0. The first two conjunctions correspond to the first two of the bracketings, but the third conjunction does not correspond to the third bracketing. Instead it corresponds to the 'flat' bracketing [$a^kb^kc^k$], which no context-free grammar can associate with the class of strings contained within the bracketing, because the latter is a context-sensitive language! On the other hand, the reason that there is no disjoint sum in I_9 corresponding to the bracketing [a^k [b^k] c^k] is the same as the absence of such a sum in I_{8m}: a^kc^k is not a substring of $a^kb^kc^k$ (k > 0).Since a^kc^k is a subsequence of $a^kb^kc^k$ for all k, each string of the form $a^kb^kc^k$ is a four-way disjoint sum in the QIS I_{9Q}: three that correspond to the three bracketings assigned by every context-free grammar to that string, plus one that corresponds to the flat bracketing.

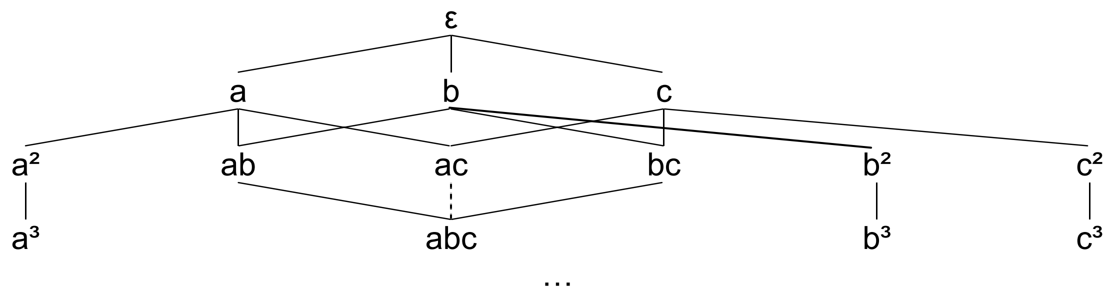

Figure 28. I_9 and I_{9Q} for the inherently ambiguous context-free language $S_9 = \{a^mb^nc^p$: m, n, p ≥ 0, m = n or m = p or n = p}

3.3. A context-free replacement for the regular language S_3 in I_3

In section 2, it was pointed out that because of the non-commutativity of concatenation, the generator set for the regular language $S_3 = \{(a \mid b)^n: n \geq 0\}$ in the SIS I_3 is identical to the entire language, i.e. that no member of S_3 can be generated by conjunction. However there is a context-free SIS $I_{3\beta} = \langle S_{3\beta}, \models\rangle$ in which $S_{3\beta}$ is obtained by replacing each member of S_3 that entails ba with a new member that entails ab and no longer entails ba, so that disjunction and conjunction are total functions, and some members of $S_{3\beta}$ are disjoint or overlapping sums, including all those in $S^{*\prime}_1$. To illustrate, compare $I_{3ab+ba} = \langle S_{3ab+ba}, \models\rangle$ in Figure 29, in which S_{3ab+ba} = {ε, a, b, ab ba}, with $I_{3\beta\textsc{b}ab} = \langle S_{3\beta\textsc{b}ab}, \models\rangle$ in Figure 30, in which $S_{3\beta\textsc{b}ab}$ = {ε, a, b, ab, $\textsc{b}a\beta$}. In the latter, the string $\textsc{b}a\beta$ replaces ba in the former, where $\textsc{b}$ is a copy of b but distinct from it, and $\beta = bb^{-1}$, in which b^{-1} is the string inverse of b, so that β (the trace of b), like ε, has zero length. Disjunction and conjunction are total functions in $S_{3\beta\textsc{b}ab}$, and its generator set $S^*_{3\beta\textsc{b}ab}$, italicized in Figure 30, is a proper subset of $S_{3\beta\textsc{b}ab}$, since ab is the disjoint sum of the pair a, b in $I_{3\beta\textsc{b}ab}$, whereas conjunction is not defined for a, b in I_{3ab+ba} and the generator set is identical to the entire set.

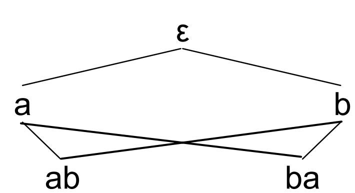

Figure 29. I_{3ab+ba}, a finite substructure of I_3 showing failure of disjunction and conjunction

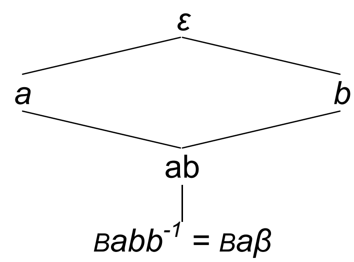

Figure 30. $I_{3\beta\mathrm{B}ab}$ for the sublanguage $S_{3\beta\mathrm{B}ab}$ of $S_{3\beta}$ in which disjunction and conjunction are total functions

The complete SIS $I_{3\beta} = \langle S_{3\beta}, \vDash \rangle$, in which $S_{3\beta} = \{x\beta^n b^p\text{: } n, p \geq 0, x \in \{(\mathrm{B}^j a^k)^i\text{: } i, j \geq 0, k > 0\}$ and $\#\mathrm{B} = n\} = \{\varepsilon, a, b, a^2, ab, \mathrm{B}a\beta, b^2, a^3, a^2b, a\mathrm{B}a\beta, ab^2, \mathrm{B}a^2\beta, \mathrm{B}a\beta b, \mathrm{B}^2a\beta^2, \ldots\}$, is defined recursively in (4).[21] Recursive step (4.b.i) defines the language $S_1 \subsetneq S_{3\beta}$, which provides input to step (4.b.ii) for specifying the remaining members of $S_{3\beta}$, in which at least one B precedes an a. These procedures together provide a recursive specification of movement as copy and deletion, in which a single b on the right edge of a string, immediately preceded by an a and zero or more traces, is deleted (i.e. replaced by a trace) and a copy of b is inserted on the left edge of the string.[22] For example, when applied to ab, (4.b.ii) yields $\mathrm{B}a\beta$, corresponding to ba in S_3; and to $a\mathrm{B}a\beta b$, it yields $\mathrm{B}a\mathrm{B}a\beta^2 = (\mathrm{B}a)^2\beta^2$, corresponding to $(ba)^2$ in S_3.

4. Recursive definition of $S_{3\beta}$
   a. Base case: $\varepsilon \in S_{3\beta}$.
   b. Recursive steps:
      i. If $s \in S_{3\beta}$, then $as \in S_{3\beta}$ and $sb \in S_{3\beta}$.
      ii. If $t = xa\beta^n b$ $(j \geq 0) \in S_{3\beta}$, then $u = \mathrm{B}xa\beta^{n+1} \in S_{3\beta}$.
   c. Closure: Nothing else is in $S_{3\beta}$.

Figure 31 diagrams $I_{3\beta} = \langle S_{3\beta}, \vDash \rangle$ for strings of length ≤ 4. The atomic chain sublanguages of $S_{3\beta}$ are the same as for S_3 and S_1, namely $X_{3\beta a} = \{a^m\text{: } m \geq 0\}$ and $X_{3\beta b} = \{b^n\text{: } n \geq 0\}$, so that the generator set $S^*_{3\beta}$ of $S_{3\beta}$, italicized in Figure 31, includes these as well as many other members of $S_{3\beta}$. The complement set $S^{*\prime}_{3\beta} = S_{3\beta} - S^*_{3\beta}$ consists of $S^{*\prime}_1 = \{a^m b^n\text{: } m, n > 0\} \cup \{a\mathrm{B}x\beta b\} \cup \{a^m\mathrm{B}x\beta b^n\text{: } m > 1 \text{ and } x \nvDash a^m, \text{ or } n > 1\}$, where $x \in S_{3\beta}$, $x \vDash a$, and $x \nvDash b$.

[21] $S_{3\beta}$ is not a regular language because the number of B's in each of its members must equal the number of β's.

[22] Discuss relevance to the trace theory of movement with REFs.

Table 1 shows some of the properties of members of $S_{3\beta}$ of length ≤ 4 excluding the two atomic chain sublanguages. The second column provides the counterparts in S_3 of the listed member of $S_{3\beta}$; and the third column the maximal conjuncts of which the member is the sum and that are not substrings of each other. If two are listed, the member is the disjoint or overlapping sum of those conjuncts; and if one is listed, it belongs to a non-atomic chain sublanguage to which the member also belongs.

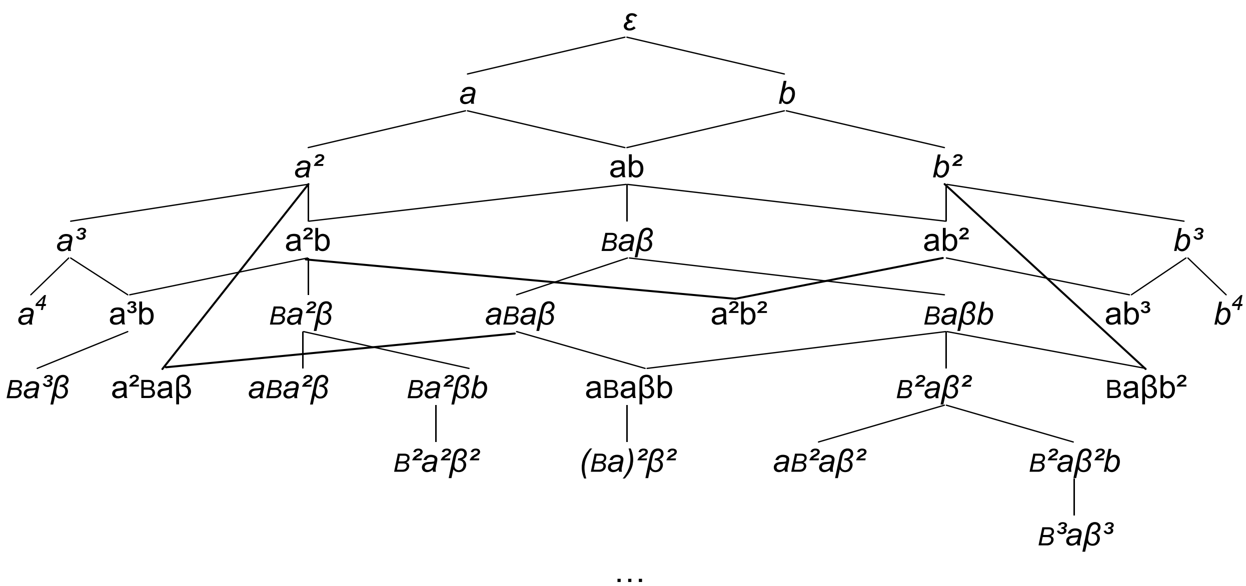

Figure 31. $I_{3\beta}$ for $S_{3\beta}$, a context-free variant of S_3 in which disjunction and conjunction are total functions

$S_{3\beta}$ members of length ≤ 4 excluding $T_{3\beta a}$, $T_{3\beta b}$	S_3 counterparts	Maximal independent substrings
ab	ab	a, b
$\mathrm{B}a\beta$	ba	ab
a^2b	a^2b	a^2, ab
ab^2	ab^2	ab, b^2
$a\mathrm{B}a\beta$	aba	$\mathrm{B}a\beta$
$\mathrm{B}a^2\beta$	ba^2	a^2b
$\mathrm{B}a\beta b$	bab	$\mathrm{B}a\beta$
$\mathrm{B}^2a\beta^2$	b^2a	$\mathrm{B}a\beta$
a^3b	a^3b	a^3, a^2b
a^2b^2	a^2b^2	a^2b, ab^2
ab^3	ab^3	ab^2, b^3
$a^2\mathrm{B}a\beta$	a^2ba	a^2, $a\mathrm{B}a\beta$
$a\mathrm{B}a^2\beta$	aba^2	$\mathrm{B}a^2\beta$
$\mathrm{B}a^3\beta$	ba^3	a^3b
$a\mathrm{B}a\beta b$	$(ab)^2$	$a\mathrm{B}a\beta$, $\mathrm{B}a\beta b$
$a\mathrm{B}^2a\beta^2$	ab^2a	$\mathrm{B}^2a\beta^2$
$\mathrm{B}a^2\beta b$	ba^2b	$\mathrm{B}a^2\beta$
$(\mathrm{B}a)^2\beta^2$	baba	$a\mathrm{B}a\beta b$

$S_{3\beta}$ members of length ≤ 4 excluding $T_{3\beta a}$, $T_{3\beta b}$	S_3 counterparts	Maximal independent substrings
$B^2a^2\beta^2$	b^2a^2	$Ba^2\beta b$
$Ba\beta b^2$	bab^2	b^2, $Ba\beta b$
$B^2a\beta^2b$	b^2ab	$Ba\beta b^2$
$B^3a\beta^3$	b^3a	$B^2a\beta^2b$

Table 1. Some properties of members of $S_{3\beta}$; italicized members are generators

3.4. How to determine whether a language in an SIS is context free

A defining property of a context-free language S is that every grammar G that generates S is center embedding, i.e. has at least one non-terminal symbol A such that A ⇒ tAv in G, where t and v are non-null terminal strings and A ⇒ u, where u is a terminal string. If A is a start symbol of G, then the strings u and s = tuv are members of S, and if u is non-null, u is a center substring of w, defined as in ().[23] Thus center embedding can give rise to center substrings in a language, but it is not the only source, since ab is a center substring of a^2b^2 in S_1, and S_1 is a regular language.

5. u ∈ S is a center substring of s ∈ S if and only if there are non-empty strings t, v such that s = tuv there are no strings x, y such that s = xu or s = uy.[24]

A logical characterization of the requirement for a language S in an SIS to be context free can be given using the notion of center substring degree, analogous to center embedding degree. A string s ∈ S has center substring degree 1 (CS° 1) with u ∈ S if u is a maximal center substring of s, i.e. if s = tuv, where t and v are non-null, and u is not a substring of any other center substring of s. CS° is defined recursively in (6).

6. For all n > 0, s has CS° n+1 with u if s = twv where w ∈ S is a maximal center substring of s and w has CS° n with u.

It now may be observed that S is a context-free language in an SIS I = <S, ⊨> if and only if S contains a chain sublanguage X with no bound on CS° for members of X with some y ∈ X. For example, the context-free language S_5 is identical to its atomic chain sublanguage X_{5ab}, in which there is no bound on CS° for members of X_{5ab} with the atom ab. In S_{5-1}, there is no bound on CS° for members of $X_{5-1a1} = \{a^nb^{n-1}: n > 0\} \cup \{a^nb^n: n \geq 0\}$ with the string ab.[25] In S_7, there is no bound on CS° for members of any chain sublanguage with the atom ac or bd. On the other hand, while there is no bound on CS° of strings of the form a^+b^+ with ab in the regular language S_1 as a whole, there is no chain sublanguage X_1 of S_1 with that property.

---

[23] If u in A ⇒ u is null, then take u to be tv and s to be t^2v^2.

[24] The requirement that there be no strings x, y such that s = xu or s = uy rules out, for example, b as a center substring of b^3.

[25] Note that $a^nb^n \to a^{n+1}b^n = a^{n+1}b^n$, not a^{n+1}, $a^{n+1}b$, …, nor $a^{n+1}b^{n+1}$, since none of the latter strings belong to S, a fact that is critical to the construction of X_{5-1a1}.

4. The calculus of context-sensitive languages

This section describes logical structures for two types of well-known mildly context-sensitive languages. REF needed First is I_{11} = <S_{11}, ⊨> in Figure 32, in which S_{11} is the context-sensitive language $\{a^nb^nc^n: n \geq 0\} = \{\varepsilon, abc, a^2b^2c^2, a^3b^3c^3, \ldots\}$. Every non-empty member of S_{11} belongs to an atomic sublanguage, so is logically independent of every other such member.[26] Second is I_{12} = <S_{12}, ⊨> in Figure 33, in which S_{12} = {xy: x ∈ $\{a^mb^n$: m, n ≥ 0}; y ∈ $\{c^md^n$: m, n ≥ 0}, the copy of x with c in place of a and d in place of b}. X_{12ac} and X_{12bd} are atomic chain sublanguages of S_{12}; every other member of S_{12} belongs to an atomic sublanguage. Like I_{12} is I_{13} = <S_{13}, ⊨> in Figure 34, in which S_{13} = {xy: x ∈ $\{(a \mid b)^n$: n ≥ 0}; y ∈ $\{(c \mid d)^n$: n ≥ 0}, the copy of x with c in place of a and d in place of b}.

Figure 32. I_{11} for the context-sensitive language $S_{11} = \{a^nb^nc^n: n \geq 0\}$

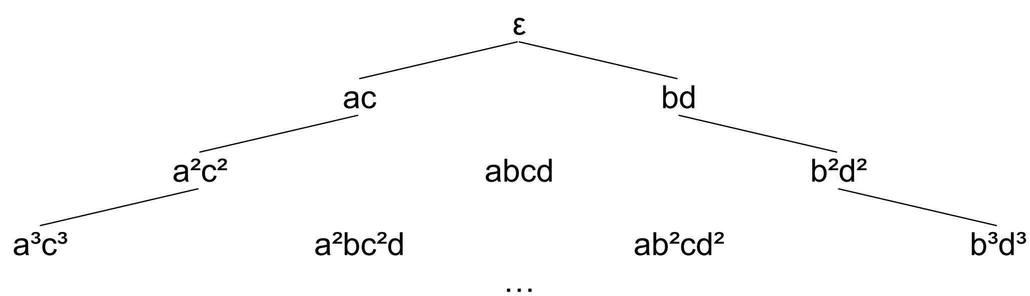

Figure 33. I_{12} for S_{12} = {xy: x ∈ $\{a^mb^n$: m, n ≥ 0}; y ∈ $\{c^md^n$: m, n > 0}, the copy of x with c in place of a and d in place of b}

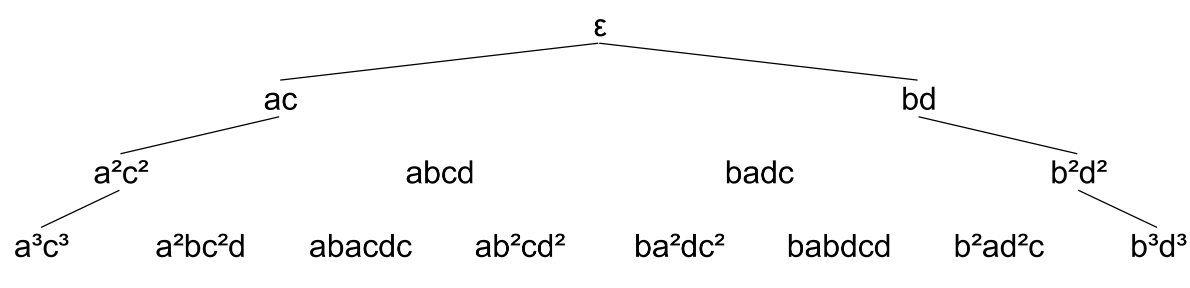

Figure 34. I_{13} for S_{13} = {xy: x ∈ $\{(a \mid b)^n$: n > 0}; y ∈ $\{(c \mid d)^n$: n > 0}, the copy of x with c in place of a and d in place of b}

These SISs for mildly context-sensitive languages all have unboundedly many members that belong to atomic sublanguages, making the choice of SIS inappropriate for logical

[26] The substring (solid) arcs connecting ε to other than its shortest superstring(s) have been omitted in Figure 32 through Figure 34.

investigation of their specific properties, but raising the possibility that it is a defining feature of a significant subclass of context-sensitive languages.[27]

4.1. *Sequence implication structures for context-sensitive languages*

QISs provide richer and potentially more useful structures for the analysis of context-sensitive languages. For example, the QIS $I_{11Q} = \langle S_{11}, \vDash_Q \rangle$ in Figure 35, is isomorphic to the SIS I_5 for the context-free language $\{a^n b^n: n \geq 0\}$ in Figure 19.[28] Moreover, the QIS $I_{12Q} = \langle S_{12}, \vDash_Q \rangle$ in Figure 36, is isomorphic to the SIS I_1 for the regular language $\{a^m b^n: m, n \geq 0\}$ in Figure 3.[29] The generator set for S_{12} in I_{12Q} is the context-free language $S^*_{12Q} = \{a^m c^m: m > 0\} \cup \{b^n d^n: n > 0\}$, italicized in Figure 36, and its complement the context-sensitive language $S^{*\prime}_{12Q} = \{xy: x \in \{a^m b^n: m, n > 0\}; y \in \{c^m d^n: m, n > 0\}$, the copy of x with c in place of a and d in place of b}. Finally, the QIS $I_{13Q} = \langle S_{13}, \vDash_Q \rangle$ in Figure 37, is isomorphic to the SIS I_3 for the regular language $S_3 = \{(a \mid b)^n: n \geq 0\}$ in Figure 6. The generator set for S_{13} in I_{13Q} is identical to S_{13} for the same reason that the generator set for S_3 is identical to S_3 in I_3.

Other applications of the use of QIS to the study of context-sensitive languages can be made, such as the investigation of inherent ambiguity in languages like $\{a^m b^n c^p d^q: m, n, p, q \geq 0; m = n = q \text{ or } m = p = q\}$.

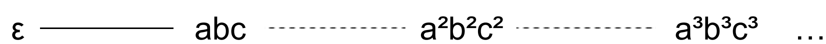

Figure 35. I_{11} and I_{11Q} for S_{11}; cf. Figure 19

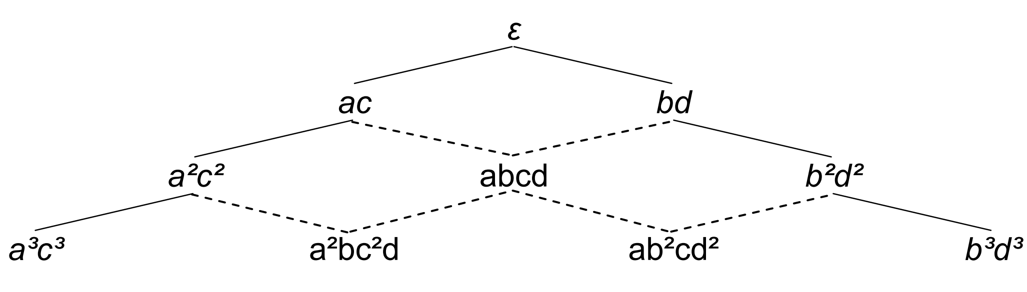

Figure 36. I_{12} and I_{12Q} for S_{12}; cf. Figure 3 and Figure 24

---

[27] Not all context-sensitive languages have this property, for example $\{a^n b^{n^2}: n > 0\} = \{ab, a^2b^4, a^3b^9, \ldots\}$, whose SIS is isomorphic to I_5.

[28] For example $a^2b^2c^2 \vDash_Q abc$ in I_{8Q}, since abc can be analyzed as r_1r_2 where $r_1 = ab$, $r_2 = c$, and $a^2b^2c^2$ as $q_0r_1q_1r_2q_2$ where $q_0 = a$, $q_1 = b$ and $q_2 = c$, so that $q_0q_1q_2 = abc \in S_8$.

[29] For example $abcd \vDash_Q ac$ in I_{9Q}, since ac can be analyzed as r_1r_2 where $r_1 = a$, $r_2 = c$, and abcd as $q_0r_1q_1r_2q_2$ where $q_0 = \varepsilon$, $q_1 = b$ and $q_2 = d$, so that $q_0q_1q_2 = bd \in S_9$.

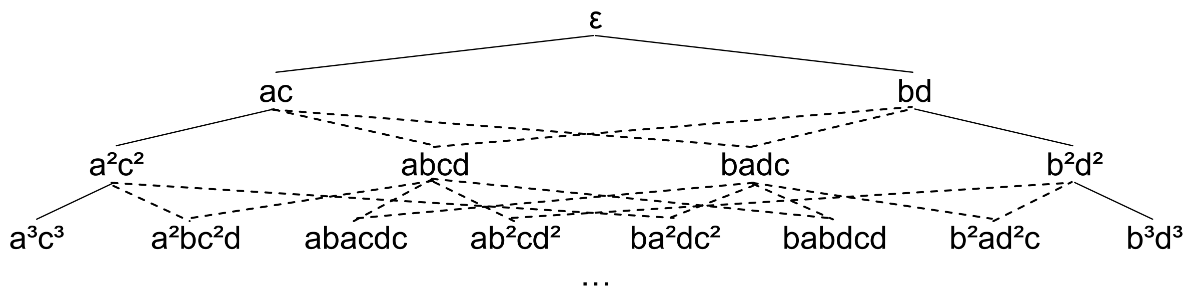

Figure 37. I_{13} and I_{13Q} for S_{13}; cf. Figure 6 and Figure 25

4.2. Context-sensitive replacements for the context-free language S_{7Q} in the QIS I_{7Q} and context-sensitive language S_{13Q} in the QIS I_{13Q}

In section 3.1, it was pointed out that because of the non-commutativity of concatenation, the generator set for the context-free mirror image language $S_7 = \{xy: x \in \{(a \mid b)^n: n \geq 0\}, y \in \{(c \mid d)^n: n \geq 0\}$, the mirror image of x with c in place of a and d in place of b} in the QIS I_{7Q} is identical to the entire language. However the context-sensitive QIS $I_{7\beta Q} = \langle S_{7\beta}, \vDash_Q \rangle$ in which $S_{7\beta}$ is obtained by replacing each member of S_7 that entails bacd with a new member that entails abdc and no longer entails bacd, analogous to the definition of the SIS $I_{3\beta}$, and with comparable results. Figure 38 represents the finite substructure $I_{7\beta Q\textsc{b}a\beta\textsc{c}d\gamma} = \langle S_{7\beta\textsc{b}a\beta\textsc{c}d\gamma}, \vDash_Q \rangle$ of $I_{7\beta Q}$, in which $S_{7\beta\textsc{b}a\beta\textsc{c}d\gamma} = \{\varepsilon, ac, bd, abdc, \textsc{b}a\beta\textsc{c}d\gamma\}$ is the sublanguage of the string $\textsc{b}a\beta\textsc{c}d\gamma$ in $S_{7\beta}$, where $\textsc{b}$ and β are as in $S_{3\beta}$, $\textsc{c}$ is a copy of c, and $\gamma = cc^{-1}$ (the trace of c). The language $S_{7\beta}$ is context sensitive, as is shown by the fact that the intersection of $S_{7\beta}$ with the regular language $\{\textsc{b}^i a\beta^j \textsc{c}^k d\gamma^m: i, j, k, m \geq 0\}$ is the context-sensitive language $\{\textsc{b}^n a\beta^n \textsc{c}^n d\gamma^n: n \geq 0\}$, and that context-sensitive languages are closed under intersection with regular languages. A similar result is obtained by replacing the QIS I_{13Q} by $I_{13\beta Q} = \langle S_{13\beta}, \vDash_Q \rangle$, in which the context-sensitive language $S_{13\beta}$ is obtained by replacing each member of S_{13} that entails badc with a new member that entails abcd and no longer entails bacd, analogous to the definition of the SIS $I_{7\beta}$, and with comparable results. Figure 39 represents the finite substructure $I_{13\beta Q\textsc{b}a\beta\textsc{d}c\delta} = \langle S_{13\beta\textsc{b}a\beta\textsc{d}c\delta}, \vDash_Q \rangle$ of $I_{13\beta Q}$, in which $S_{13\beta\textsc{b}a\beta\textsc{d}c\delta} = \{\varepsilon, ac, bd, abdc, \textsc{b}a\beta\textsc{d}c\delta\}$ is the sublanguage of the string $\textsc{b}a\beta\textsc{d}c\delta$ in $S_{13\beta}$, where $\textsc{b}$ and β are as in $S_{3\beta}$, $\textsc{d}$ is a copy of d, and $\delta = dd^{-1}$ (the trace of d).

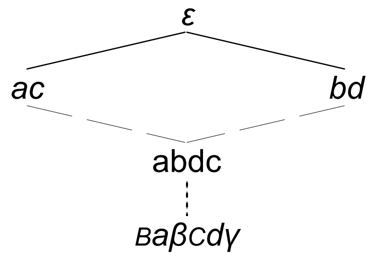

Figure 38. $\mathbf{I_{7\beta Q_{Ba\beta cd\gamma}}}$ **for the sublanguage** $\mathbf{S_{7\beta_{Ba\beta cd\gamma}}}$ **of** $\mathbf{S_{7\beta}}$

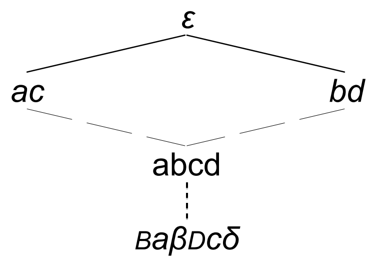

Figure 39. $\mathbf{I_{13\beta Q_{Ba\beta Dc\delta}}}$ **for the sublanguage** $\mathbf{S_{13\beta_{Ba\beta Dc\delta}}}$ **of** $\mathbf{S_{13\beta}}$

5. Applications for the study of natural languages

Not yet written.

References

REFs needed for trace theory of movement and for mildly context-sensitive languages.

Chomsky, Noam. 1963. Formal properties of grammars. In R. Duncan Luce, Robert R. Bush and Eugene Galanter, eds., *Handbook of Mathematical Psychology*, vol. II, pp. 323-418. New York: John Wiley and Sons.

Ferré, Sébastien. 2007. The efficient computation of complete and concise substring scales with suffix trees. In S. O. Kuznetsov and S. Schmidt, eds., *Formal Concept Analysis* (*Lecture Notes in Computer Science* 4390), pp. 98-113. Berlin: Springer.

Koslow, Arnold. 1992. *A Structuralist Theory of Logic*. Cambridge: Cambridge University Press.

Langendoen, D. Terence. 2002. Sequence structure. In Bruce Nevin & Stephen M. Johnson, eds., *The Legacy of Zellig Harris: Language and Information into the 21st Century*, vol. 2: *Computability of Language and Computer Applications*, pp. 61-75. Amsterdam: John Benjamins.

Leonard, Henry and Nelson Goodman. 1938. The calculus of individuals. *Journal of Symbolic Logic*.

Parikh, Rohit. 1961. Language generating devices. *Research Laboratory of Electronics Quarterly Progress Report* 60: 199-212.