Incremental Processing and Acceptability

Glyn Morrill

Report LSI-98-46-R
Incremental Processing and Acceptability

Glyn Morrill*
Universitat Politècnica de Catalunya

We present a left to right incremental algorithm for the processing of Lambek categorial grammar by proof net construction. A simple metric of complexity, the profile in time of the number of unresolved valencies, correctly predicts a wide variety of performance phenomena: garden pathing, left to right quantifier scope preference, centre embedding unacceptability, preference for lower attachment, and heavy noun phrase shift.

Introduction

Contemporary linguistics rests on abstractions and idealizations which, however fruitful, must eventually be integrated in a mature cognitive science with human computational performance in language use. In this paper we consider the case of language processing on the basis of Lambek categorial grammar (Lambek 1958). We argue that an incremental algorithm of proof net construction affords an account of various processing phenomena, including:

(1)

- garden pathing
- left to right quantifier scope preference
- centre embedding unacceptability

Garden pathing (Bever 1970) is illustrated by the following contrasts:

(2)

a. The horse raced past the barn.
   b. ?The horse raced past the barn fell.

(3)

a. The boat floated down the river.
   b. ?The boat floated down the river sank.

(4)

a. The dog that knew the cat disappeared.
   b. ?The dog that knew the cat disappeared was rescued.

Typically, although the ‘b’ sentences are perfectly well-formed they are perceived of as being ungrammatical due to a strong tendency to interpret their initial segments as in the ‘a’ sentences.

Left to right quantifier scope preference is illustrated by:

(5)

a. Someone loves everyone.
   b. Everyone is loved by someone.

Both sentences exhibit both quantifier scopings:

* Department de Llenguatges i Sistemes Informàtics, Mòdul C 5 - Campus Nord, Jordi Girona Salgado 1-3, E-08034 Barcelona. E-mail: morrill@lsi.upc.es; http://www.lsi.upc.es/~glyn/.
(6) a. $\exists x \forall y (\text{love } x, y)$
    b. $\forall y \exists x (\text{love } x, y)$

However, while the dominant reading of (5a) is (6a), that of (5b) is (6b), i.e. the preference is for the first quantifier to have wider scope. Note that the same effect is observed when the nature of the quantifications is swapped:

(7) a. Everyone loves someone.
    b. Someone is loved by everyone.

While both sentences in (7) have both quantifier scoping, the preferred readings give the first quantifier wide scope.

Centre embedding unacceptability is illustrated by the fact that while the nested subject relativizations of (8) exhibit little variation in acceptability, the nested object relativizations (9) exhibit a sever deterioration in acceptability (Chomsky 1965, ch. 1).

(8) a. The dog that chased the cat barked.
    b. The dog that chased the cat that saw the rat barked.
    c. The dog that chased the cat that saw the rat that ate the cheese barked.

(9) a. The cheese that the rat ate stank.
    b. The cheese that the rat that the cat saw ate stank.
    c. The cheese that the rat that the cat that the dog chased saw ate stank.

We argue that a single simple metric of categorial processing complexity explains these and other performance phenomena.

1 Lambeck calculus

We shall assume some familiarity with Lambeck categorial grammar as presented in e.g. Moortgat (1988), Morrill (1994), Moortgat (1997) or Carpenter (1998), and limit ourselves here to reviewing some technical and computational aspects.

The types, or (category) formulas, of Lambeck calculus are freely generated from a set of primitives by binary infix connectives / ("over"), \ ("under") (directional divisions) and \* ("product"). With respect to a semigroup algebra $(L, +)$ (i.e. a set $L$ closed under an associative binary operation + of adjunction) each formula $A$ is interpreted as a subset $\llbracket A \rrbracket$ of $L$ by residuation as follows.

\[
\begin{align*}
\llbracket A \cdot B \rrbracket &= \{ s_1 + s_2 | s_1 \in \llbracket A \rrbracket \land s_2 \in \llbracket B \rrbracket \} \\
\llbracket A \backslash B \rrbracket &= \{ s|s' \in \llbracket A \rrbracket, s' + s \in \llbracket B \rrbracket \} \\
\llbracket B / A \rrbracket &= \{ s|s' \in \llbracket A \rrbracket, s + s' \in \llbracket B \rrbracket \}
\end{align*}
\]

A sequent, $\Gamma \Rightarrow A$, comprises a succedent formula $A$ and an antecedent configuration $\Gamma$ which is a a finite sequence of formulas.\(^1\) A sequent is valid if and only if in all interpretations the ordered adjunction of elements inhabiting the antecedent formulas always yields an element inhabiting the succedent formula. The following Gentzen-style sequent presentation is sound and complete for this interpretation (Buszkowski 1986, Došen 1992), and indeed for free semigroups (Pentus 1994); hence the Lambeck calculus

\[^1\] Officially the antecedent is non-empty, a detail we gloss over.
can make an impressive claim to be the logic of concatenation; the parenthetical notation \( \Gamma(\Delta) \) represents a configuration containing a distinguished subconfiguration \( \Delta \).

(11)  
\[ \begin{align*}  
\text{a. } & A \Rightarrow A \quad \text{id} \quad \Gamma \Rightarrow A \quad \Delta(A) \Rightarrow B \quad \text{Cut} \\
& \frac{\Delta(\Gamma) \Rightarrow B}{A, \Gamma \Rightarrow B} \quad \text{L} \\
& \frac{A \Rightarrow \Gamma \Rightarrow B}{\Gamma \Rightarrow A \backslash B} \quad \text{R} \\
\text{b. } & \Gamma \Rightarrow A \quad \Delta(B) \Rightarrow C \quad \Delta(\Gamma, A \backslash B) \Rightarrow C \\
& \frac{\Gamma, A \Rightarrow B}{A, \Gamma \Rightarrow B} \quad \text{L} \\
& \frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow B / A} \quad \text{R} \\
\text{c. } & \Gamma \Rightarrow A \quad \Delta(B) \Rightarrow C \quad \Delta(B/A, \Gamma) \Rightarrow C \\
& \frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow B / A} \quad \text{L} \\
& \frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow B / A} \quad \text{R} \\
\text{d. } & \Gamma(A, B) \Rightarrow C \quad \Gamma \Rightarrow A \quad \Delta \Rightarrow B \\
& \frac{\Gamma(A \cdot B) \Rightarrow C}{\Gamma, \Delta \Rightarrow A \cdot B} \quad \text{L} \\
& \frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \cdot B} \quad \text{R} \\
\end{align*} \]

By way of example, "lifting" \( A \Rightarrow B / (A \backslash B) \) is generated as follows:

(12)  
\[ \begin{align*}  
& \frac{A \Rightarrow A \quad B \Rightarrow B}{A \Rightarrow (A \backslash B) \Rightarrow B} \quad \text{L} \\
& \frac{A \Rightarrow B}{A \Rightarrow B / (A \backslash B)} \quad \text{R} \\
\end{align*} \]

And "composition" \( A \backslash B, B \backslash C \Rightarrow A \backslash C \) is thus:

(13)  
\[ \begin{align*}  
& \frac{B \Rightarrow B \quad C \Rightarrow C}{A \Rightarrow A \quad B, B \backslash C \Rightarrow C} \quad \text{L} \\
& \frac{A, A \backslash B, B \backslash C \Rightarrow C}{A \backslash B, B \backslash C \Rightarrow A \backslash C} \quad \text{R} \\
\end{align*} \]

Every rule with the exception of Cut, where the Cut formula \( A \) does not appear in the conclusion, has exactly one connective occurrence less in its premisses than in its conclusion; Lambek (1958) proved Cut elimination — that every proof has a Cut-free counterpart — hence a decision procedure for theoremhood is given by backward chaining proof search in the Cut-free calculus. The non-atomic instances of the id axiom are derivable from atomic instances by the rules for the connectives. But even in the Cut-free atomic-id calculus there is spurious ambiguity: equivalent derivations differing only in irrelevant rule ordering.

One approach to this problem consists in defining normal form derivations in which the succession of rule application is regulated (König 1989, Hepple 1990; see e.g. Hendriks 1993 ch. 3). Each sequent has a distinguished category formula (underlined) on which rule applications are keyed:

(14)  
\[ \begin{align*}  
\text{a. } & P \Rightarrow P \quad \text{id} \\
& \frac{\Delta(A) \Rightarrow C}{\Delta(\Gamma, A \backslash B) \Rightarrow C} \quad \text{L} \\
& \frac{A, \Gamma \Rightarrow B}{\Gamma \Rightarrow A \backslash B} \quad \text{R} \\
\end{align*} \]
c. \[ \Gamma \Rightarrow A \quad \Delta(B) \Rightarrow C \]
\[\Delta(B/A, \Gamma) \Rightarrow C \quad /L \]
\[\Gamma, A \Rightarrow B \quad /R \]
\[\Gamma \Rightarrow B/A \]

d. \[\Gamma \Rightarrow A \quad \Delta \Rightarrow B \]
\[\Gamma, \Delta \Rightarrow A: B \quad /R \]

e. \[\Gamma(A) \Rightarrow B \]
\[\Gamma(A) \Rightarrow B \]

In the regulated calculus there is no spurious ambiguity, and provided there is no explicit or implicit antecedent product, i.e. provided -L is not needed, \( \Gamma \Rightarrow A \) is a theorem of the Lambek calculus iff \( \Gamma \Rightarrow A \) is a theorem of the regulated calculus. However, apart from the issue regarding -L, there is a general cause for dissatisfaction with this approach: it assumes the initial presence of the entire sequent to be proved, i.e. it is in principle non-incremental; on the other hand, allowing incrementality on the basis of Cut would reinstate with avengance the problem of spurious ambiguity: for then what are to be the Cut formulas? Consequently, the sequent approach is ill-equipped to address the basic asymmetry of language, the asymmetry of its processing in time, and has never been forwarded in a model of the kind of processing phenomena cited in the introduction.

An alternative, combinatory, presentation would comprise combinatory schemata such as the following (together with a Cut rule, feeding one rule application into another):

(15)

a. \( A, A \backslash B \Rightarrow B \quad B/A, A \Rightarrow B \)

b. \( A \Rightarrow (B/A) \backslash B \quad A \Rightarrow B/(A\backslash B) \)

c. \( A \backslash B, B \backslash C \Rightarrow A \backslash C \quad C/B, B/A \Rightarrow C/A \)

By a result of Zielonka (1981) the Lambek calculus is not axiomatizable by any finite set of combinatory schemata, so no such combinatory presentation can constitute the logic of concatenation in the sense of Lambek calculus. Combinatory categorial grammar (Steedman 1997) does not concern itself with the capture of all (or only) the concatenatively valid combinatory schemata, but from its inception (Ades and Steedman 1981) it has emphasized the capacity of schemata such as those of (15) to produce left-branching, and therefore incrementally processable, analyses, e.g. on a shift-reduce design. An approach, also based on regulation of the succession of rule application, to the associated problem of spurious ambiguity is given in Hepple and Morrill (1989) but again, to our knowledge, there is no predictive relation between incremental combinatory processing and the kind of processing phenomena cited in the introduction.

2 Proof nets

Lambek categorial derivations are usually presented in the style of natural deduction or sequent calculus. Here we concern ourselves with categorial proof nets (Roorda 1991) as the fundamental structures of proof in categorial logic, in the same sense that linear
proof nets were originally introduced by Girard (1987) as the fundamental structures of proof in linear logic. (Cut-free) proof nets exhibit no spurious ambiguity and play the role in categorial grammar that parse trees play in phrase structure grammar.

A polar category formula is a Lambek categorial type labelled with input (°) or output (\(^ *)\) polarity. A polar category formula tree is a binary ordered tree in which the leaves are labelled with polar atoms (literals) and each local tree is one of the following (logical) links:

\[
\begin{align*}
\text{a. } & A^o \quad B^* \quad A^* \quad B^o \\
& \xrightarrow{1} \quad A \setminus B^* \quad A \setminus B^o
\end{align*}
\]

\[
\begin{align*}
\text{b. } & B^* \quad A^o \quad A^* \quad B^o \\
& \xrightarrow{2} \quad B \setminus A^* \quad B \setminus A^o
\end{align*}
\]

\[
\begin{align*}
\text{c. } & A^* \quad B^* \quad B^o \quad A^o \\
& \xrightarrow{3} \quad A \setminus B^* \quad A \setminus B^o
\end{align*}
\]

Without polarities, a formula tree is a kind of construction tree of the formula at its root: daughters are labelled with the immediate subformulas of their mothers. The polarities indicate sequent sidedness, input for antecedent and output for succedent; the polarity propagation follows the sidedness of subformulas in the sequent rules: in the antecedent (input) rule for \(A \setminus B\) the subformula \(A\) goes in a succedent (output) and the subformula \(B\) goes in an antecedent (input), in the succedent (output) rule for \(A \setminus B\) the subformula \(A\) goes in an antecedent (input) and the subformula \(B\) goes in a succedent (output), etc. The labels \(i\) and \(ii\) indicate whether the corresponding sequent rule is unary or binary. Note that in the output links the order of the subformulas is switched; this corresponds to a cyclic reading of sequents: the succedent type is adjacent to the first antecedent type.

A proof frame is a finite sequence of polar category formula trees exactly one of which has a root of output polarity (corresponding to the unique succedent of sequents).

An axiom linking on a set of literal labelled leaves is a partitioning of the set into pairs of complementary leaves which is planar in their ordering, i.e. there are no two pairs \(\{L_1, L_3\}, \{L_2, L_4\}\) such that \(L_1 < L_2 < L_3 < L_4\). Geometrically, planarity means that where the leaves are ordered on a line, paired leaves can be connected in the half plane without crossing. Axiom links correspond to id instances in a sequent proof.

A proof structure is a proof frame together with an axiom linking on its leaves. A proof net is a proof structure in which every elementary (i.e. visiting vertices at most once) cycle crosses the edges of some i-link. Geometrically, an elementary cycle is the perimeter of a face or cluster of faces in a planar proof structure. There is a proof net with roots \(A^*, A_1^*, \ldots, A_n^*\) iff \(A_1, \ldots, A_n \Rightarrow A\) is a valid sequent.

---

2 This criterion, adapted from that of Lecomte and Ratoré (1995), derives from Girard's (1987) 'long trip condition', which is a highly involved mathematical result. Danos and Regnier (1989) express it in terms of acyclicity and connectivity of certain subgraphs. Intuitively, acyclicity ensures that the subformulas of i-links (binary rules) occur in different subproofs, whereas connectivity ensures that the subformulas of i-links (unary rules) occur in the same subproofs (attributed to Jean Gallier by Philippe de Groote, p.c.). However the single-succedent (intuitionistic) nature of Cut-free categorial proof in fact renders the connectivity requirement redundant, hence we have just an acyclicity test.
Let us assume the following lexical assignments:

(17)  

<table>
<thead>
<tr>
<th>barn</th>
<th>:=</th>
<th>barn</th>
</tr>
</thead>
<tbody>
<tr>
<td>:=</td>
<td></td>
<td>CN</td>
</tr>
<tr>
<td>horse</td>
<td>:=</td>
<td>horse</td>
</tr>
<tr>
<td>:=</td>
<td></td>
<td>CN</td>
</tr>
<tr>
<td>past</td>
<td>:=</td>
<td>(\lambda x.\lambda y.\lambda z.(\text{past} \ x \ (y \ z)))</td>
</tr>
<tr>
<td>:=</td>
<td></td>
<td>(((N\backslash S)\backslash (N\backslash S))/N)</td>
</tr>
<tr>
<td>raced</td>
<td>:=</td>
<td>race</td>
</tr>
<tr>
<td>:=</td>
<td></td>
<td>(N\backslash S+)</td>
</tr>
<tr>
<td>the</td>
<td>:=</td>
<td>(N/CN)</td>
</tr>
</tbody>
</table>

The feature + on S marks the projection of a tensed verb form; a verb phrase modified by "past" need not be tensed. Let us consider the incremental processing of (2a) as proof net construction. We assume initially that an S is expected; after perception of the word "the" there is the following partial proof net (for simplicity we omit features, included in lexical entries, from proof nets):

(18)

Here there are three unmatched valencies/unordered dependencies; no axiom links can yet be placed, but after "horse" we can build:

(19)

Now there are only two unmatched valencies. After "raced" we have, on the correct analysis, the following:

(20)

Note that linking the Ns is possible, but we are interested in the history of the correct
analysis, and in that the verb valencies are matched by the adverb that follows (henceforth we indicate only the principal connective of a mother node):

(21)

Observe that a cycle is created, but as required it crosses the edges of a i-link. At the penultimate step we have:

(22)

The final proof net analysis is given in figure 1.

The semantics associated with a categorial proof net, i.e. the proof as a lambda term (intuitionistic natural deduction proof, under the Curry-Howard correspondence) is extracted by associating a distinct index with each output division node and travelling as
follows, starting by going up at the unique output root (de Groote and Retoré 1996):

(23) 
- travelling up at the mother of an output division link, 
  perform the lambda abstraction with respect to the 
  associated index of the result of travelling up at the 
  daughter of output polarity;

- travelling up at the mother of an output product link, 
  form the ordered pair of the result of travelling up at 
  the right daughter (first component) and the left 
  daughter (second component);

- travelling up at one end of an axiom link, continue 
  down at the other end;

- travelling down at an (input) daughter of an input 
  division link, perform the functional application of 
  the result of travelling down at the mother to the 
  result of travelling up at the other (output) daughter;

- travelling down at the left (resp. right) daughter of an 
  input product link, take the first (resp. second) 
  projection of the result of travelling down at the 
  mother;

- travelling down at the (input) daughter of an output 
  division link, return the associated index;

- travelling down at a root, return the associated lexical 
  semantics.

Thus for our example we obtain (24a), which is logically equivalent to (24b).
(24) a. \(\lambda x\lambda y\lambda z(past\: x\: (y\: z))\: (the\: barn)\: \lambda 1(race\: 1)\: (the\: horse))\)
   b. \((past\: (the\: barn)\: (race\: (the\: horse)))\)

The analysis of (2b) is less straightforward. Whereas in (2a) "raced" expresses a one-place predication ("go quickly"), in (2b) it expresses a two-place predication (there was some agent racing the horse); "horse" is modified by an agentless passive participle, but the adverbial "past the barn" is modifying "race". Within the confines of the Lambek calculus the characterization we offer assumes the lexical assignment to the passive participle given in the following.\(^3\)

(25) fell - fall
    := N\( \setminus S^+ \)
   raced - \((\lambda x\lambda y\lambda z((y\: z) \land \exists w(x\: z\: w)),\: race2)\)
    := ((C(N\( \setminus CN\))/N\( \setminus S^-\))-N\( \setminus S^-\)))

Here "raced" is classified as the product of an untensed transitive verbal type, which can be modified by the adverbial "past the barn" by composition, and an adnominalizer of this transitive verbal type. According to this, (2b) has the proof net analysis given in figure 2. The semantics extracted is (26a), equivalent to (26b)

(26) a. \((fall\: (the\: (\pi_1(\lambda x\lambda y\lambda z((y\: z) \land \exists w(x\: z\: w)),\: race2)\: \lambda 29\: \lambda 30
   \lambda u\lambda v\lambda w(past\: u\: (v\: w))\: (the\: barn)\: \lambda 41((
   (\pi_2(\lambda r\lambda s\lambda t((s\: t) \land \exists q(p\: q)),\: race2\: 29\: 41\: 30)\: horse))))\)
   b. \((fall\: (the\: \lambda 8((horse\: 8) \land \exists 7(past\: (the\: barn)\: (race2\: 8\: 7)))))\)

Let us assign to each proof net analysis a complexity profile which indicates, before and after each word, the number of unmatched literals, i.e. unresolved valencies or dependencies, following the processing up to that point. This is a measure of the course of memory load in optimal incremental processing. We are not concerned here with resolution of lexical ambiguity or serial backtracking; we are supposing sufficient resources that the non-determinism of selection of lexical entries and their parallel consideration is not the critical burden. Rather, the question is: which among parallel competing analyses places the least load on memory?

The complexity profile is easily read off a completed proof net: the complexity between two words is the number of axiom links bridging rightwards (forwards in time) at that point. Thus for (2a) and (2b) analysed in figures 1 and 2 the complexity profiles are as follows:

(27)

\[
\begin{array}{cccccc}
6 & \ & b \\
5 & \ & \ & \ & \ & \ \\
4 & \ & \ & \ & \ & \ \\
3 & \ & ab & b & b & b \\
2 & \ & ab & a & a & b \\
1 & \ & \ & a & a & b \\
0 & \ & \ & \ & \ & \ \\
\end{array}
\]

a. the horse raced past the barn
b. the horse raced past the barn fell

We see that after the first words the complexity of the correct analysis of (2b) is con-

\(^3\) In general grammar requires the expressivity of more powerful categorial logics than just Lambek calculus; however, so far as we are aware, the characterizations we offer within the Lambek calculus bear the same properties with regard to our processing considerations as their more sophisticated categorial logic refinements, because the latter concern principally generalisations of word order, whereas the semantic dependencies on which our complexity metric depends remain the same.
sistent higher than that of its ‘garden path’ (2a), just as would be expected on the assumption that in (2b) the less costly but incorrect analysis is assiallant.

Let us consider now quantifier scope preference. A rudimentary account of sentence-peripheral quantifier phrase scoping is obtained in Lambek calculus by means of lexical assignments such as the following:

\[
\begin{align*}
\text{everyone} & \quad \lambda x \forall y(x \ y) \quad ::= \quad St/(N\setminus St) \\
\text{everyone} & \quad \lambda x \exists y(x \ y) \quad ::= \quad (St/N\setminus St) \\
\text{someone} & \quad \lambda x \exists y(x \ y) \quad ::= \quad St/(N\setminus St) \\
\text{someone} & \quad \lambda x \exists y(x \ y) \quad ::= \quad (St/N\setminus St)
\end{align*}
\]

Then one analysis of (5a) is that given in figure 3. This is the subject wide scope analysis: its extracted and simplified semantics is as in (29).

\[
\begin{align*}
\text{(29) a.} & \quad (\lambda x \exists y(x \ y) \ \lambda 1(\lambda x \forall y(x \ y) \ \lambda 2(\text{love} \ 2 \ 1))) \\
\text{b.} & \quad \exists x \forall y(\text{love} \ y \ x)
\end{align*}
\]

A second analysis is that given in figure 4. This is the object wide scope analysis: its extracted and simplified semantics is as in (30).

\[
\begin{align*}
\text{(30) a.} & \quad (\lambda x \forall y(x \ y) \ \lambda 2(\lambda x \exists y(x \ y) \ \lambda 1(\text{love} \ 2 \ 1))) \\
\text{b.} & \quad \forall y \exists x(\text{love} \ y \ x)
\end{align*}
\]

Let us compare the complexity profiles for the two readings:

\[
\begin{array}{c|c|c|c|c}
\text{by} & \text{is} & \text{loved} \\
\hline
\text{ab} & \text{ab} & \text{ab} \\
\text{ab} & \text{ab} & \text{ab} \\
\end{array}
\]

A \forall \exists analysis of (5b) is given in figure 5. This has semantics, after some simplification, (33), which is equivalent to (30).

\[
\text{(33) } \forall x \exists y \exists z[[9 = 7] \land (\text{love} \ 16 \ 9)]
\]
An $\exists v$ analysis of (5b) is given in figure 6. This has semantics, after some simplification, (34), which is equivalent to (29).

(34) \[
\exists v 1437[7 = 16 \land (\text{love} \ 14 \ 7)]
\]

Again, the preferred reading has the lower complexity profile:

(35)

\[
\begin{array}{cccc}
6 & b & b & a \\
5 & b & b & \textbf{ab} \\
4 & a & a & \textbf{ab} \\
3 & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
2 & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
1 & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
0 & \textbf{ab} & \textbf{ab} & \textbf{ab}
\end{array}
\]

a. everyone is loved by someone $\forall v$ (figure 5)

b. everyone is loved by someone $\exists v$ (figure 6)

Turning now to the final performance phenomenon listed in the introduction, for subject and object relativisation we assume the relative pronoun lexical assignements (36).

(36)

\[
\begin{aligned}
\text{that} & \quad \lambda x \lambda y \lambda z [(y \ z) \land (x \ z)] \\
& \quad \text{(CN} \setminus \text{CN})/(\text{S} \setminus \text{CN})
\end{aligned}
\]

\[
\begin{aligned}
\text{that} & \quad \lambda x \lambda y \lambda z [(y \ z) \land (x \ z)] \\
& \quad \text{(CN} \setminus \text{CN})/(\text{S} \setminus \text{CN})
\end{aligned}
\]

Sentence (8b) is analysed in figure 7. Sentence (9b) is analysed in figure 8. Let us compare the complexities:

(37)

\[
\begin{array}{cccccc}
9 & b & b & b & \textbf{ab} & \textbf{ab} \\
8 & b & b & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
7 & b & b & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
6 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
5 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
4 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
3 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
2 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
1 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} \\
0 & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab} & \textbf{ab}
\end{array}
\]

a. the dog that chased the cat that saw the rat barked

b. the cheese that the rat that the cat saw ate stank

Again, the profile of (9b) is higher; indeed it rises above 7–8, thus reaching what are usually taken to be the limits of short term memory.
Figure 2
"the horse raced past the barn fell"
Figure 3
"someone loves everyone" ($\exists\forall$)

Figure 4
"someone loves everyone" ($\forall\exists$)
"everyone is loved by someone" (∃v)
Figure 8
"the cheese that the rat that the cat saw ate stank"
3 Further cases

Another dramatic example of unacceptability is provided by the following:

(38)  a. That two plus two equals four surprised Jack.
    b. *That two plus two equals four surprised Jack astonished Ingrid.*
    c. *That two plus two equals four surprised Jack astonished Ingrid bothered Frank.*

The passive parafrases, however, seem more or less equally acceptable:

(39)  a. Jack was surprised that two plus two equals four.
    b. Ingrid was astonished that Jack was surprised that two plus two equals four.
    c. Frank was bothered that Ingrid was astonished that Jack was surprised that two plus two equals four.

In figure 9 we give the analysis of (38b) and in figure 10 that of (39b). We now abbreviate proof nets by flattening formula trees into their linear representations; since this conceals the order switching of output links the notation belies the underlying planarity. It is very interesting to observe that the complexity profile of the latter is in general lower even though the analysis has more than twice the total number of links.

By way of another example, Kimball (1973, p.27) observes that in a sentence such as (41), three ways ambiguous according to the attachment of the adverb, the lower the attachment is, the higher the preference (what he terms 'Right Association')

(41) Joe said that Martha believed that Ingrid fell today.

In figure 11 we give the analyses for the highest, the middle, and the lowest attachments. Accordingly, the complexity profiles are:

The same effect occurs strongly in (43), where the preferred reading is the one given by the lowest attachment, even though that one is the nonsensical reading.

(43) the book that shocked Mary's title
"that that two plus two equals four surprised Jack astonished Ingrid"

The analyses are given in figure 12. The complexities are thus:
Finally, our account appears to explain the preference for heavy noun phrases to appear at the end of the verb phrase (heavy noun phrase shift). Of the following the second is more acceptable:

(45) a. John gave the painting that Mary hated to Bill.
    b. John gave Bill the painting that Mary hated.

The analyses are given in figure 13. The complexities are thus:

(46) a. John gave the painting that Mary hated to Bill.
    b. John gave Bill the painting that Mary hated.

References


Figure 10: "Angela was astonished that Jack was surprised that two plus two equals four."
Figure 11

"Joe said that Martha believed that Ingrid fell today."

Joe said that Martha believed that Ingrid fell today.
Figure 12
"the book that shocked Mary's title" (sensical and nonsensical)
Figure 13
"John gave the painting that Mary hated to Bill" vs. "John gave Bill the painting that Mary hated"


LSI-98-4-R “BayesProfile: application of Bayesian Networks to website user tracking”, Ramón Sangüesa and Ulises Cortés.


LSI-98-7-R “Trust Values for Agent Selection in Multiagent Systems”, Karmelo Urzelai.

LSI-98-8-R “The use of SAREL to control the correspondence between Specification Documents”, Núria Castell and Àngels Hernández.

LSI-98-9-R “Intervalizing colored graphs is NP-complete for caterpillars with hair length 2”, C. Álvarez, J. Díaz and M. Serna.


LSI-98-12-R “Height-relaxed AVL rebalancing: A unified, fine-grained approach to concurrent dictionaries”, Luc Bougé, Joaquim Gabarro, Xavier Meseguer and Nicolas Schabanel.

LSI-98-13-R “HyperChromatic trees: a fine-grained approach to distributed algorithms on RedBlack trees”, Xavier Meseguer and Borja Valles.


LSI-98-20-R "Visualization of Cerebral Blood Vessels", Anna Puig.


LSI-98-23-R "Incorporating the Behavioural Information to the Schema Construction Process of Federated Data Bases System", Luis Carlos Rodríguez G.


LSI-98-25-R "Construcción automática de diccionarios de patrones de extracción de información", Neus Català and Núria Castell.

LSI-98-26-R "Syntactic Connectivity", Glyn Morrill.

LSI-98-27-R "Geometric Distance Constraint Satisfaction by Constraint-to-constraint relaxation", Lluis Solano Albajes and Pere Brunet Crossa.


LSI-98-29-R "Computing Directional Constrained Delaunay Triangulations", Marc Vigo Anglada.

LSI-98-30-R "On the complexity of moving vertices in a graph", Antoni Lozano (Universitat Politècnica de Catalunya) and Vijay Raghavan (Vanderbilt University).


LSI-98-33-R "Fuzzy Heterogeneous Neurons for Imprecise Classification Problems", Julio J. Valdés, Lluís A. Belanche, René Alquézar.


Hardcopies of reports can be ordered from:

Núria Sánchez
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Campus Nord, Mòdul C6
Jordi Girona Salgado, 1-3
08034 Barcelona, Spain
secrelsi@lsi.upc.es

See also the Department WWW pages, http://www-lsi.upc.es/