On Partially Ordered Trees in Syntactic Description

In Honor of Ivan Sag[1]

Edward L. Keenan 4/25/2013

In current Minimalist approaches to syntactic description it is often assumed that the (labeled) trees used to represent syntactic structure exist independently of the expressions whose structure they are being used to represent. Thus many nodes in the tree may be "unfilled" initially in a derivation and filled in later by moving fragments (or subtrees) of the original expression into them.

The trees represent two kinds of ordering information: the hierarchical ordering (dominance) relations in terms of which constituents are defined, the linear order of terminal expressions and the constituents that contain them. In addition (not our concern here) a labeling function assigns grammatical category names to non-terminal nodes in a tree and lexical expressions (roughly) to terminal nodes. Our concern here is with an apparently unmotivated, restrictive assumption concerning the relation between hierarchical order and the linear order. We begin with a straightforward definition of a (finite) tree that does not make this assumption and illustrate two advantages it has: (1) it can represent directly certain kinds of discontinuous constituents without affecting semantic interpretation, and (2) it allows a representation of morphology as directly structural, not simply a recoding of hierarchical structure (and labeling).

Definition 1

a. A (*linguistic*) *tree* T is a pair (N,D), where N is a finite set whose elements are called *nodes*, D is a binary relation on N called *dominates*, satisfying:
   1. D is a partial order of N (i.e. D is reflexive, antisymmetric and transitive) We say a node x *strictly* dominates (SD) a node y iff xDy and $x \neq y$. A node x is called a *leaf* iff there is no node y such that xSDy.
   2. There is an $r \in N$ such that for all $y \in N$, rDy. (r is provably unique and called the *root* of (N,D); it follows, trivially, that N is non-empty.)
   3. For any node x, $\{y \in N \mid yDx\}$ is a chain: [uDx & vDx] ⇒ [uDv or vDu].

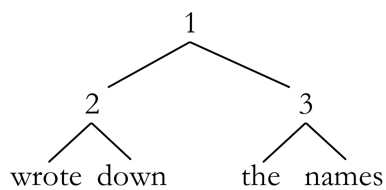
b. A (*partially*) *ordered* tree is a triple T = (N,D,<), where (N,D) is a tree and < is a strict (irreflexive) linear order of the leaves of (N,D). '<' is read (*strictly*) *precedes*. We extend < to a relation <* on N (not just the leaf nodes in N) by setting: x <* y iff every leaf node that x dominates precedes every leaf node that y dominates. It follows from our definitions that if one of x and y dominates the other then neither can precede other. But if neither dominates the other it does not follow that one precedes (<*) the other. We return to this point below.

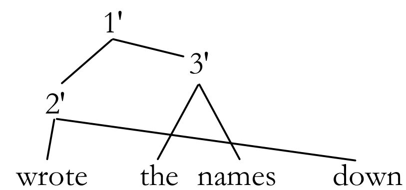
Definition 2 For each node x of a partially ordered tree T, $\downarrow x =_{df} \{y \in N \mid xDy)$. We write T[x] for that subtree of T whose dominance relation is the dominance relation of T restricted to ↓x and whose linear order relation on its leaf nodes is, again, just that of T restricted to those in ↓x. T[x] is provably a tree with root x. The *constituents* of T are just {T[x] | x a node of T}.

Definition 3 Given trees T and T', T ≃ T' (T is *isomorphic* to T') iff there is a bijection h from N to N' such that x dominates y in T iff h(x) dominates h(y) in T', and, x precedes y in T iff h(x) precedes h(y) in T'.

Standardly, isomorphic trees share all of their structurally definable properties – e.g. ones definable in terms of dominance and precedes.

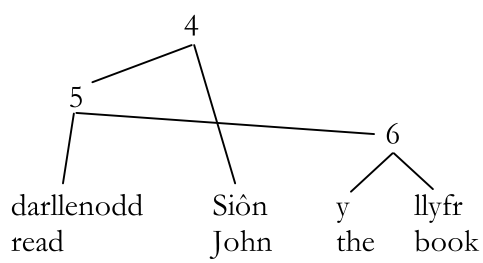
Now, assuming some standard conventions, observe that T[1] and T[1'] below are both partially ordered trees. They have the same number of nodes, 7, hence there are 7! = 5,040 bijections from ↓1 to ↓1', but none are isomorphisms as none preserve <. For example 2 <* 3 in T[1] but no non-leaf node stands in the <* to any non-leaf node in T[1']. But the immediate constituents T[2], T[3] of T[1] are isomorphic to those of T[1'], namely T[2'] and T[3'] respectively.





Note that assuming Compositionality – that the semantic interpretation of an expression is a function of that of its immediate constituents it follows that T[1] and T[1'] have the same interpretation since, writing denotations-in-a-model in boldface, they are the same function of interpretations of the same expressions, **wrote down** and **the names**. Moreover what holds intralinguistically above holds cross linguistically as well. Compare (1) from Welsh with the expected tree for its English translation (in which the verb and object are adjacent, but V and the object form a constituent in both English and Welsh using partially ordered trees):

(1)



John read the book

Again the English and the Welsh sentences are interpreted identically assuming the lexical items are correctly translated.

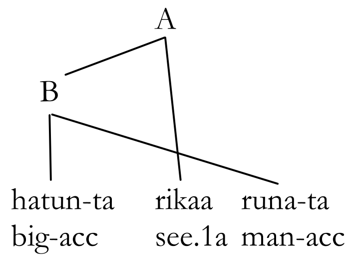
We'll want to consider objections to using partially ordered trees but first let us expand slightly the empirical range of phenomena we can represent. Several languages, notably Warlpiri, Quechua and Latin, can use matching morphology on non-adjacent expressions to indicate their semantic constituenthood. From Quechua (Huallaga), example from Weber (1989) first; structure of (2b) shown in (3):

(2) a. [[hatun runa]-ta rikaa (Constituency as bracketed)
big man -acc see.1s
I see the big man

b. hatun-ta rikaa runa-ta
big-acc see.1s man-acc
I see the big man

c. runa-ta rikaa hatun-ta
man-acc see.1s big-acc
I see the big man

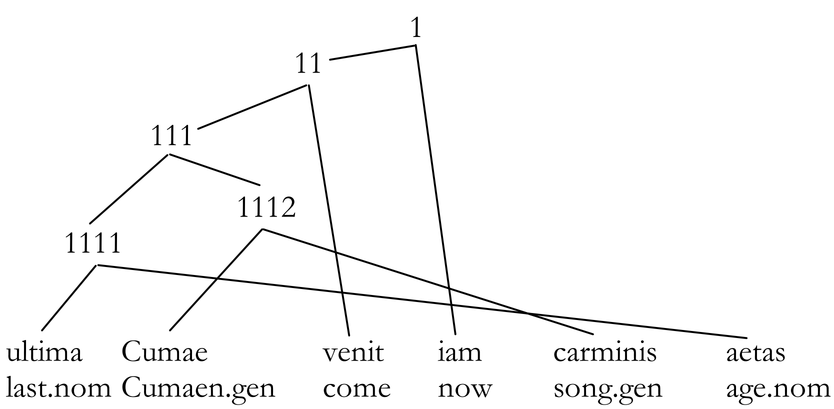
(3)



We note that when the adjective and the noun it modifies are adjacent we only have one accusative marker, but both noun and adjective take the marker when they are separated. In that case they form a semantically interpreted constituent, and it is the matching morphology which tells us that. So in (2b,c) we have not destroyed constituent structure, we have just recoded it in the bound morphology. In (2b)/(3) and (2c) *hatun-ta* and *runa-ta* form a constituent which is a sister to *rikaa* 'I see'. The branching structure is identical to that in (1) up to labeling (not considered).

A last example, taken from Matthews (1981) from Latin shows that matching case endings (shape conditioned by the category of the expression) enable us to identify widely discontinuous constituents.

(4)



The last age of the Cumean song has now arrived

Why do linguists not use trees as we have defined them, allowing certain kinds of discontinuous constituents? The formal definition is rigorous and clear. An historical answer is that in the early days of generative grammar we had no explicit mechanisms of semantic interpretation but assumed that expressions derived from the same underlying (“deep”) structure had the same meaning. So if we derived *wrote the names down* from *wrote down the names* we at least accounted for their sameness of meaning. But once explicit interpretative mechanisms became available we can show that independently derivable expressions are logical paraphrases. For example *Ted doesn’t know every student in the class* is logically equivalent to *There is at least one student in the class that Ted doesn’t know*. But we are not motived to syntactically derive either of these sentences from the other. (And within sentential or first order logic each sentence is logically equivalent to infinitely many others).

However the assumption of meaning determined on underlying structures and preserved by transformations hasn’t been part of generative grammar for well over a generation. When I have pushed linguists on why discontinuous constituents are somehow banned from initial position in derivations I have occasionally received embarassingly naive answers – “We don’t countenance crossing lines in tree diagrams”. This is purely notational – like banning the use of blue ink to draw trees, or banning wide margins. There is of course a less silly way to state what appears to be intended – it is just that it is so obviously false. Namely, “To interpret two (or more) expressions as a semantic unit they must be adjacent”. Formally we can guarantee this by imposing the following additional condition on the class of trees we allow:

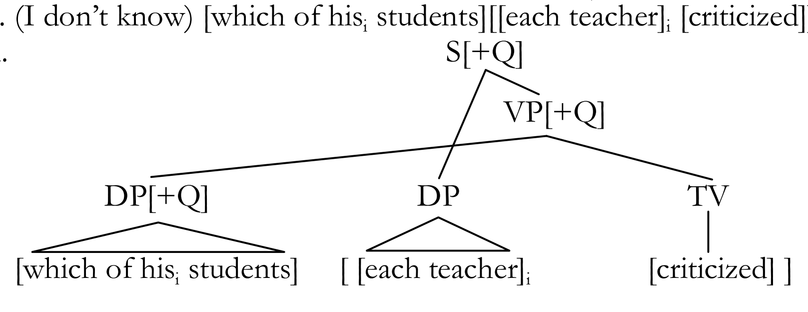
(5) ***Totality***: for all nodes x,y if neither x nor y dominate the other then either x <* y or y <* x.

So Totality guarantees that non-leaf nodes are linearly ordered to the extent possible (they can’t be ordered recall if they are in a dominance relation).

But why should we impose Totality? It just seems obviously false for the four examples we have given. And worse, it forces us to ignore a basic part of linguistic structure – the trade off between word order and morphological marking. It is a very old observation that overt case marking allows greater word order freedom. This says that morphology should be part of the structure of an expression that is semantically interpreted. Not imposing Totality allows this in a natural way.

I close with a last case, due to Jim Blevins (1994), which touches on issues of more direct concern to modern generative grammar. We observe first that binding, as indicated by subscripts, is natural in (6a) but not in (6b), which suggests to many that a binder must c-command a pronominal it binds. (Constituency indicated by bracketing here). But then on standardish assumptions binding should be out in (6c), but it isn’t. If we assign it the rough structure in (6d) we see that the binder still c-commands the bindee.

(6) a. [each teacher]$_i$ [criticized [many of his$_i$ students]]
b. [many of his$_i$ students][criticized [each teacher]$_j$] ($i \neq j$)
c. (I don’t know) [which of his$_i$ students][[each teacher]$_i$ [criticized]]
d.



[1] This article is related to Ivan's work by what I hope is its clarity and rigor. I do not know if Ivan ever concerned himself directly with issues I address here.

References

1. Blevins, James. 1994. Derived constituent order in unbounded dependency constructions. *J. of Linguistics* **30**:349–409.
2. Matthews, Peter. 1981. *Syntax*. Cambridge University Press.
3. Weber, David John. 1989. *A Grammar of Huallaga Quechua*. UC Press.