The Evolution of Model-Theoretic Frameworks in Linguistics*

Geoffrey K. Pullum

School of Philosophy, Psychology and Language Sciences
University of Edinburgh
Edinburgh EH8 9LL, Scotland, UK
pullum@gmail.com

1. Introduction The varieties of mathematical basis for formalizing linguistic theories are more diverse than is commonly realized. For example, the later work of Zellig Harris might well suggest a formalization in terms of CATEGORY THEORY, since Harris takes the utterances of a language to constitute a collection (possibly not even enumerable) that is not itself of particular interest, and concentrates on discussing a set of associative, composable mappings defined on it. And thoroughgoing versions of generative semantics like Pieter Seuren's seem to call for a formalization in terms of TRANSDUCERS, mapping meanings to phonetic forms and conversely. However, work in formal syntax over the past fifty years has been entirely dominated by just one kind of mathematics: the string-manipulating combinatorial systems categorized as **generative-enumerative syntax** (henceforth GES) in Pullum and Scholz (2001).

A GES grammar is a recursive definition of a specific set, *ipso facto* computably enumerable.[1] The definition is given in one of two ways: top-down or bottom-up. The classic top-down style of GES grammar is a program for a nondeterministic process of construction by successive expansion symbol strings. It amounts in effect to a program with the property that if it were left running forever, choosing randomly but always differently among the possibilities allowed for expanding symbols, every possible string in the desired set would ultimately get constructed. Such a device is described as GENERATING the entire set of all structures the machine is capable of constructing. The production systems developed by Post (1943), developed in an effort to mathematicize the syntactic approach to logical deduction, are of this type. So are all the familiar types of phrase structure grammar, classic MIT transformational grammar (TG), Backus-Naur form, and all typical statements of the syntax of programming languages.

The other type of GES grammar consists of a finite set of primitive elements (typically a set of lexical items) and a finite set of operations for composing them into larger complex units. Such a system is taken to define the set of expressions obtained by closing the set of primitives under the combination operations. This type covers all of categorial grammar including Montagovian syntax, tree adjoining grammar, the 'minimalist program', the neo-minimalism of Stabler and Keenan, and nearly all statements of the formation rules for logics.

2. MTS frameworks I want to try and show how the type of theoretical framework that is becoming known as MODEL-THEORETIC SYNTAX (MTS) was actually adumbrated as long ago as 1970 or even before, and a small number of linguists advocated it more explicitly by 1980, but proper mathematical development did not begin until the 1990s. But first I will sketch the hallmarks of MTS frameworks as Rogers, Scholz, and I understand them, and then take a close look at certain earlier works that represented halting steps toward MTS.

I use the term EXPRESSIONS for sentences, clauses, phrases, words, idioms, lexemes, syllables — the objects that linguists investigate. I take these to *have* syntactic structure, not merely to be analyzable in terms of structures imposed on them or posited for them by linguists. That is, I take a realist view of expressions and of their syntactic properties.

MTS frameworks, as I understand them, are distinguished by the adoption of three general positions: (I) rules are statements about expressions; (II) grammars are finite sets of such rules; (III) well-formedness of an expression consists in satisfaction of the grammar. Each of these points needs a little more discussion.

3. Rules MTS rules are simply assertions about the structure of expressions. That is, an MTS rule makes a statement that is either true or false when evaluated in the

---

[1]Following Soare (1996), I consider words like 'recursive' and 'recursion' to be undesirably ambiguous, and prefer 'computably enumerable' over 'recursively enumerable'.

structure of an expression. If a structure is to be grammatically well formed according to a certain rule, then the rule must be true as interpreted in that structure.

Rules within GES are not like this. A GES rule is an instruction forming part of a procedure for stepwise construction of a DERIVATION — a rule-mediated sequence of representations, the last of which is by definition well formed. Crucially, GES rules do not assert anything about the structure of well-formed expressions; they are instructions making up individual parts of an integrated procedure for building such structures, and they cannot be interpreted in isolation.

Nowhere is this clearer than in recent TG. 'Merge' cannot be understood as a condition on the structure of expressions. It is a dynamic tree-building concatenation operation, joining two items together and adding a node immediately dominating them. Notice that it is stated informally as an imperative. The same is true for 'Move α' in early TG: it is an instruction forming part of a non-deterministic random generation procedure, permitting a constituent of type α to shift to some other location at the next stage in the derivation.

The same thing is true for phrase structure rules, however. The rule 'PP → P NP' does not state that adpositions precede NPs. If the grammar contained a rule 'PP → NP P' in addition, then adpositions would be freely ordered. If it contained a rule 'P → e', there might be no adpositions in the generated expressions at all. Everything depends on the combined functions of the component parts of a grammar holistically defining a set.

MTS rules, by contrast, are naturally given informal statement as declarative clauses. Examples might be 'The subject noun phrase of a tensed clause is in the nominative case'; 'The head verb of the verb phrase in a tensed clause agrees in person and number with the subject of that clause'; 'Verbs always follow their direct objects'; or 'Attributive modifiers precede the heads that they modify'.

4. Grammars   An MTS grammar is simply a finite, unordered set of MTS rules. This means that individual rules in grammars can be developed and assessed piecemeal, without regard to any sequencing of applications.

For example, how exactly to frame the general statement of verb agreement can proceed independently of how to state the conditions on auxiliary selection in passives or positioning of relative pronouns in relative clauses. No condition on structure overrides or takes priority over another such condition. The conditions all have to be true in an expression structure if it is to count as well formed. A linguist stating a grammatical rule need only be attentive to what expressions there are and what structures they have — nothing about sequencing of operations or stages of construction is relevant.

Grammar, on the MTS view, is about *what structure expressions have*. It is not about devising a sequence of operations that would permits the construction of the entire set of all and only those structures that are grammatical.

5. Grammaticality   An expression is well formed according to an MTS grammar if and only if the semantic consequences of the grammar are true in its syntactic structure. Grammaticality is thus defined by reference to the SEMANTIC consequences of rules (the semantics of the formal language in which the rules are stated, that is — not the semantics of the natural language being described). An expression is fully well formed if and only if its structure complies with every requirement that is a semantic consequence of what the grammar says.

Thus a rule saying 'every direct object noun phrase in a transitive verb phrase immediately follows the verb' is satisfied only by structures in which every transitive verb phrase containing a direct object noun phrase does indeed have that noun phrase adjacent to and immediately following the verb. (The echo of Tarski's definition of truth is not just an allusion, of course; we are actually using Tarski's notion of a model here.)

The rule is vacuously true in an intransitive clause: where there is no object, there is nothing to falsify a statement fixing the positioning of objects. Ungrammaticality on the MTS view is defined by violation of one or more of the rules of the grammar.

6. MTS and GES   Thus far, I have been trying to clarify the notion of an MTS description, but not to claim that MTS descriptions are inherently superior to non-MTS ones. They could turn out to be entirely inadequate. Linguists have made many proposals for rules or principles that that are simply impossible to state in MTS terms. For anyone who accepts these, MTS is simply untenable. One general class of examples is that MTS does not permit statement of generalizations that demand quantification over all the expressions in a language. So MTS forbids all of these:

(i) the 'optionality' claim in X-bar theory that non-head constituents are always optional (see Kornai and Pullum 1990 and Pullum and Scholz 2001 for discussion);

(ii) the 'exhaustive constant partial ordering' claim (that any ordering restriction imposed on sibling constituents in a natural language must be the same under any parent node, regardless of its label; see Gazdar and Pullum 1981);

(iii) any 'ambiguity avoidance' constraint that bars structures on the basis of their being confusable with others (Pullum and Scholz 2001 discuss a putative Russian example);

(iv) any 'blocking' constraint that bars structures on the basis that other items take priority over them;

(v) any 'economy' claim that legitimates structures by reference to claims about alternatives being less economical.

Economy conditions, in particular, have been prominent in recent versions of GES. If any valid condition of this sort were ineliminably connected to properties that could only be stated through comparison of one structure's properties with another's, MTS would not allow for the proper description of natural language syntax at all. My belief is that not a single constraint of this sort is genuinely convincing as a part of syntax. But let there be no doubt about the fact that if there were one, MTS would have to be dismissed.

In a sense, though, MTS is founded on a very traditional idea: that a grammar should describe the syntactic structure of expressions of a language by making general statements about their syntactic properties. The rules stated in traditional grammars are of just this kind — statements imposing conditions on individual grammatical structures. And the grammatical expressions are simply those of which all the entailments of the grammar's statements are true.

Traditional grammars have been denigrated by linguists throughout most of the last century, in part because of extraneous concerns (like alleged prescriptivism) and in part because they are not explicit — their statements are not precisely stated in a formal language invented for the purpose and equipped with a denotational semantics. But the alleged failings of traditional grammar do not have to do with the idea of rules as statements about structure, or that satisfaction of the conditions is the determinant of well-formedness.

Chomsky (1962: 539) offers a revisionist view, stating that "a grammar must contain . . . a 'syntactic component' that generates an infinite number of strings representing grammatical sentences," and calls such a view "the classical model for grammar." This is misleading at best. There is nothing classical about the idea that grammars should be axiomatic systems for generating infinite sets of strings. It was under ten years old when he wrote, and represented a radical break with all previous conceptions of grammar (Pullum and Scholz 2005; Scholz and Pullum 2007). Although the organic connection of GES systems to the American structuralist descriptive linguistics of the 20th century is clear, they contrast sharply with the earlier tradition of grammatical scholarship and pedagogy.

And interestingly, within about ten years after the idea of stating at least some grammatical principles as statements about expression structure began to creep back into GES.

7. Derivations and trees   McCawley (1968), in the context of discussing certain issues about the 'base component' in TG, raised certain doubts about whether phrase structure rules should be interpreted as rewriting instructions on strings. His paper is well known, and it has been taken to represent some kind of early adoption of the MTS point of view. I will argue that it really does not, except in an indirect way. But it does bring up some interesting and relevant issues.
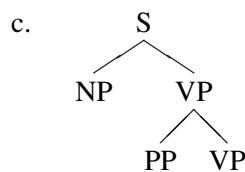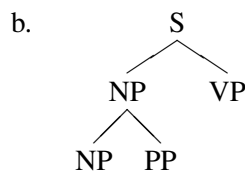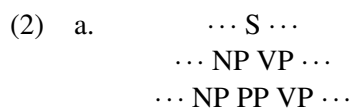
As defined in 1960s TG, a rule $A \to BC$ is a rewriting operation, permitting A to be replaced by string BC in a derivation, turning a string XAY into a string $XBCY$. A separate tree-building procedure (sketched by McCawley 1968: 245 and Lasnik 2000: 17–23) is supposed to build a parse tree from the derivation. It works from the bottom up and from the outside in, adding edges between symbols in a given line to identical symbols in the line above, and connecting up residual symbols to available nodes above. From a string ABC, a rule $B \to DE$ would produce $ADEC$, and the tree-building procedure would produce this intermediate stage:

(1)   A   B   C
       | / \ |
       A   D E C
       | | | |
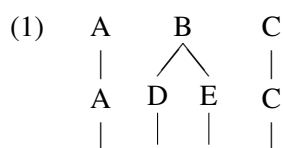
The idea is that a tree can be constructed by completing this procedure, working outside-in and bottom to top, and reducing edges like '—A—A—' to '—A—'.

But the procedure faces a problem, briefly noted by Chomsky (1959: 144, n. 8), and explored in more detail by McCawley: for some derivations it does not determine a unique tree. Worse, it may produce a tree with a structure that the rules, under their intuitive understanding, do not permit, and collapsing distinct syntactic representations.

This failure to determine a unique tree stems from the fact that a derivation — the sequence of lines resulting from the rule applications — records too little information about what the rule applications were. The derivation is a record of the content of the successive lines, with no indication of which symbol was rewritten, or which rule applied, at any given stage. For example, from the partial derivation in (2a) the standard procedure will allow either (2b) or (2c) to be built.

(2)   a.   ··· S ···
                  ··· NP VP ···
                ··· NP PP VP ···

       b.   S
                     /   \
                   NP   VP
                  / \
                NP   PP

       c.   S
                     /   \
                   NP   VP
                         / \
                       PP   VP

(Consider two possible structures for *Dogs at least bark*, one implying that there might also be other animals that

bark, the other implying that there might also be other things that dogs do.)

The point is not that the problem is insoluble under the rewriting interpretation (it is not: stipulating that the symbol rewritten must always be the leftmost one for which rewriting is possible at that point permits inference of symbol was rewritten to create the current line, which permits unique correct trees to be constructed). But McCawley was interested in how trees could be more directly answerable to the content of the phrase structure rules without any such restriction on derivation construction procedures, making the connection between rules and structures directly, through a reinterpretation of phrase structure rules. This involved taking trees to be mathematically specifiable objects in themselves, rather than just diagrammatic ways of depicting properties of equivalence classes of derivations, as in Chomsky's early work (see Chomsky 1975, pp. 181ff). In this he was perhaps influenced by the work of Zwicky and Isard (1963), sent out on December 3, 1963, to a select distribution list at the MITRE Corporation, in which a set of axioms for labeled ordered trees was given and several equivalent ways of abstractly representing trees were discussed.

McCawley considered two new possibilities for interpreting of phrase structure rules. The one I will be concerned with here was suggested to him by Richard Stanley in 1965. The idea is to interpret phrase structure rules as NODE ADMISSIBILITY CONDITIONS (henceforth, NACs). An NAC is a sufficient condition for admissibility of a node given its daughter sequence. A whole tree is to be defined as well formed iff every node in it is admissible. Under this interpretation, the rule in (3a) would be understood as informally expressed in (3b):[2]

(3) a. S → NP VP

    b. The node being evaluated is labeled 'S'; its first child is labeled 'NP'; its second child is labeled 'VP'; and there are no other child nodes.

This proposal interprets rules as monadic predicates of nodes. But it is not a proposal for MTS grammars. To see this, note that it does not respect *any* of the tenets (I) – (III).

It does not endorse (I), which says rules state necessary conditions on well-formedness of expression structures. NACs are not even defined for trees, and do not express necessary conditions anyway. Each NAC states a *sufficient* condition for admissibility of a single *node*. In consequence, (III) also fails to hold: well-formedness of a tree does not result from satisfaction of all (or in fact

[2]McCawley actually introduces a special notation, '⟨S; NP VP⟩', for a phrase structure rule with this content under the Stanley interpretation, and Gazdar (1982)140 introduces another one, '[$_S$ NP VP]'. This is unnecessary, of course. Rule statements do not have to wear their interpretations on their typographical sleeves.

any) of the NACs. And strictly, the proposal in its original context did not accept (II) either: although McCawley remarks that "node admissibility conditions are by nature unordered" (p. 248), he envisions NACs in a larger context, that of providing the deep structure trees to be the inputs to the transformations, so a grammar as a whole was not envisaged as just an unordered set of NACs.

One remark McCawley makes in connection with how NACs describe trees seems to be an error: he says that "the admissibility of a tree is defined in terms of the admissibility of all of its nodes, i.e., in the form of a condition which has the form of a logical conjunction" (p. 248). It is true that each well-formed k-node tree T will be a model of a conjunction $C_{i_1}(n_1) \land C_{i_2}(n_2) \land \cdots \land C_{i_k}(n_k)$, where i_j is the NAC that admits the node n_j, the intuitive meaning being 'node n_1 is admissible according to NAC number i_1 and node n_2 is admissible according to NAC number $i_2 \ldots$' and so on. But it is a different statement for each tree, with a number of conjuncts corresponding to the tree size. This does not yield a general definition of well-formedness according to a grammar. Note that McCawley certainly cannot have been referring to any logical conjunction of NACs, since the conjunction of two or more distinct NACs is never true at *any* node.

The correct general definition of the set of trees defined as well formed according to a given set of NACs is in fact a DISjunction. An NAC is really a one-place predicate of nodes. For example, the NAC corresponding to the rule 'A → B C' might be expressed as (4), where M and $\prec$ are interpreted by the 'mother of' and 'precedes' relations.

$$(4) \quad \varphi(x) = \begin{aligned}&(A(x)) \land \\ &((\exists y)(\exists z)[(M(x,y)) \land (M(x,z)) \land \\ &(y \prec x) \land (B(y)) \land (C(z))])\end{aligned}$$

Let $\varphi_1, \ldots, \varphi_k$ be a set of such NACs. Then the formula that would need be true in a tree to make it well formed, adding the other two assumptions of McCawley's concerning the root and leaf nodes, will be (5), where x ranges over the node set of the tree using $S(x)$ to mean that x has the designated start symbol as its label and $T(x)$ to mean that x is labeled with a member of the terminal vocabulary:

(5) $(\forall x)[\,((\text{Root}(x)) \land S(x)) \lor$ [a]

    $((\text{Leaf}(x)) \land T(x)) \lor$ [b]

    $(\bigvee_{1 \le i \le k} \varphi_i(x))\,]$ [c]

Every node [a] is the root and labeled with the start symbol, or [b] is a leaf node and labeled with a terminal symbol, or [c] satisfies the disjunction of all the NACs.

Now, the set containing just this one statement (for a given $\varphi_1, \ldots, \varphi_k$) would be a very simple example of an MTS grammar: it is a finite set of statements that may

or may not be satisfied by a given tree (trivially: it is a singleton).

However, it is in fact a grammar illustrating a description language of extremely low expressive power. It interprets NACs as primitive propositions each asserting admissibility for some specific local tree. There is really no important difference between the NAC and the local tree that it uniquely describes: a grammar could be given in the form of a finite list of local trees, the interpretation being that a tree is well formed iff it is entirely composed of local trees that are on the list. This is in fact the non-standard way of defining context-free grammars that Jim Rogers proposed at the original MTS workshop in 1996; see Rogers (1999).

As pointed out in Rogers (1997b), such a way of defining a set of trees is exactly analogous to a bigram description of a set of strings. A bigram description over an alphabet Σ is a finite list of 2-symbol sequences, and a string is grammatical according to it if every length-2 substring of the string is on the list.

But bigram descriptions define only a very small and primitive class of stringsets, the SL_2 stringsets. Local tree descriptions have much greater expressive power: every context-free stringset is the string yield of some local tree set, and every local tree set has a context-free string yield.

What McCawley apparently did not appreciate (the relevant results were not available) was that descriptions might just as well be given in a richer and more flexible description language, since no increase in weak generative capacity results from using full first-order logic on trees rather than just local tree descriptions. In fact if weak monadic second-order logic (wMSO) is used, by a crucial result later obtained by Doner (1970), a stringset defined as the string yield of the set of trees satisfying some wMSO formula is always context-free, and all context-free stringsets are thus definable.

Note that the power of first-order logic on trees is sufficient to guarantee the presence of a 'trace' in some sub-constituent accompanying a dislocated element, without using the GPSG device of having a chain of 'slashed categories' labeling all the nodes on the path between them. For example, to require that every constituent α contain exactly one node with the label β, we could say (writing dom(x,y) for 'x dominates y'):

(6) $(\forall x)[(\alpha(x)) \rightarrow ((\exists y)[\mathrm{dom}(x,y) \wedge \beta(y) \wedge$
$\qquad (\forall z)[(\mathrm{dom}(x,z) \wedge \beta(z)) \rightarrow (z=y)]])]$

The succinctness gain from the use of quantificational logic rather than just sets of NACs can be not just linear or polynomial but exponential. Jim Rogers provides the following example. Consider how to describe just a set of local trees in which the root must have m children each labeled with a distinct symbol from a list $\{A_1, \cdots, A_m\}$, any order of those children being permissible. A first-order

definition needs only to say that for each of the m labels there is exactly one child with that label. That can be done with a formula of a length linearly related to m: the formula has to say for each node x that there is an A_i such that $[A_i(x) \wedge (\forall y)[(y \not\approx x) \Rightarrow \neg(A_i(x))$ (where $1 \leq i \leq m$). But the number of distinct local trees involved, and hence the size of a grammar composed of NACs, grows as an exponential function of m (linearly related to $m!$, in fact).

In short, the Stanley/McCawley proposal for reinterpreting phrase structure rules is a very interesting idea, but its role as a precursor of MTS should not be overstated, because virtually none of the MTS program is implicit in what McCawley actually suggested.

8. Tree sequence models   Just two or three years after McCawley's paper we find the earliest published work in linguistics that can be said to adopt all three of the hallmarks of MTS. Lakoff (1969) and the more accessible Lakoff (1971), no doubt influenced by McCawley's paper, were well known and much-cited papers, and presented a radical departure from the standard way to formalize syntactic theory. But there was essentially no effect on subsequent work. The reasons are many, but it has to be said that Lakoff's ideas were ill-delineated and sloppily illustrated. If his proposals would work at all, which is doubtful, it would apparently have been of vitiatingly unrestrained expressive power.

Lakoff's reformulation of TG was put forward in the course of a defense of the generative semantics hypothesis. To remain close to the standard assumptions of the time concerning the content of syntactic description, the syntactic structures Lakoff posited were finite sequences of finite trees, exactly as in TG. Most linguists at the time thought of transformations intuitively as operations applying to trees and producing modified trees. This was not the way Chomsky had initially formalized transformations, but it proved by far the most intuitive way to think of them. Thus a set of transformations, together with various principles governing the order and domain of their application, would determine for a given tree the structural properties of the next one in the derivation. The central challenge in Lakoff's reformulation was to be able to represent what the transformational rules express concerning which trees can follow which trees in a syntactically permissible sequence.

Lakoff proposed that transformations should be stated simply as conditions on pairs of successive trees in a sequence. He remarks (using the term 'phrase-markers' for trees):

> Since transformations define possible derivations only by constraining pairs of successive phrase-markers, I will refer to transformations as 'local derivational constraints'. (Lakoff 1971: 233)

Lakoff defines a local derivational constraint as a conjunction of two statements C_1 and C_2, "where C_1 and C_2 are tree-conditions defining the class of input trees and class of output trees, respectively" (1971: 233).

The problem is that fixing the properties shared by the input trees and the properties shared by the output trees cannot possibly suffice to mimic the effect of a transformation. Chomsky (1972) briefly points this out, but the point is developed more fully in the only detailed critical study of Lakoff's proposals that I am aware of, Soames (1974). It is not necessary to go into matters of fine detail to see what the problem is. A conjunction of conditions saying anything remotely like what Lakoff suggests — that the input tree meets condition C_1 and the output tree meets condition C_2 — will associate *any* tree satisfying C_1 with *every* tree satisfying C_2.

As Chomsky notes (1972: 121, n. 19), Lakoff's formulation "would not, for example, distinguish between the identity transformation and a permutation of two nodes of the same category". That is, a transformation that would derive *They spoke about it to the others* from *They spoke to the others about it* could not be distinguished from a transformation that simply maps a structure containing two PPs to itself. This gives Lakoff unacknowledged technical problems like how to block infinite derivations of finite strings from meeting the definition of well-formedness, and how to define 'completed derivation'.

The problems run much deeper than that. Lakoff's reformulation of transformations does not guarantee conservativism, in the sense that it does not prevent wholesale change of lexical content in a derivational step. By allowing a tree representing *This, I believe* as an optional transform of *I believe this*, a grammar would also allow infinitely many other trees, with terminal strings like *That, they rejected* or *The others, we keep in the bathroom*, to count as well.

What is missing is what is guaranteed by the carry-over of already-written structure in the stepwise construction of transformational derivations: that a tree is rewritten in a way that alters it only in one specific way at each transformational step. The structures Lakoff actually desires to capture are characterized by a default which amounts to a massive redundancy: each tree is identical with the one preceding it in the sequence, except with regard to one part (typically quite small) where there is a specified change. (Lakoff may be acknowledging this when he remarks that one part of the condition C_1, repeated in C_2, "defines the part of the tree-condition which characterizes both" of two adjacent trees. But perceiving that something needs to be done is not the same as doing it.)

Thompson (1975) makes an attempt to work out Lakoff's ideas in more detail. Specifically, he aims to characterize the principle of CYCLIC APPLICATION in

terms compatible with Lakoff's proposals. (The fact that Lakoff offers no way to express the cyclic principle is noted by Soames 1974, p. 122, n. 6.)

Thompson, referring to the framework as CORRESPONDENCE GRAMMAR, assumes that each pair of adjacent trees in a well-formed sequence must be explicitly licensed by what he calls a VALIDATING RULE. He recognizes the need "to prevent extraneous changes from occurring in the derivation" — that is, to block random differences between trees and their immediate predecessors or successors that the validating rule says nothing about — so he states a global *ceteris paribus* condition on tree sequences. He assumes that for any two finite trees a finite statement of their node-by-node differences can be stated in unique form, and so will I; call this the DIFFERENCE SET for the two trees. Thompson's own statement of the condition is partly procedural, but repairing that we can restate it thus:

(7)   For each pair of adjacent trees $\langle T_{i-1}, T_i \rangle$ licensed by a validating rule R_i, any tree T_i' that is the result of eliminating from T_i some subset D of their difference set, making T_i' more similar to T_{i-1}, is such that R_i does not license the pair $\langle T_{i-1}, T_i' \rangle$.

This says that if the second tree in an adjacent pair were altered in any way that made it more similar to the first, the validating rule for the pair would no longer apply.

I note in passing that this "explicit way of seeing to it that 'everything else remains the same'" (Thompson, p. 597) yields an exponential explosion of complexity in the problem of checking a model for compliance with a grammar. Verifying this for a given pair of trees involves checking NON-satisfaction of the relevant validating rule for a set of pairs of trees of cardinality related to the power set of the difference set. The number of subsets of a difference set of size d will be 2^d, so it must be established for each of 2^d tree pairs that the validating rule fails to license them. And this must be done for the entire set of pairs in the derivation.

Let me also note that although Thompson's ultimate goal was to exhibit a "formal statement" of the principle of cyclic application, what he actually offers is a highly informal statement in English that is not a condition on structures at all, but a condition on validating rules, and almost certainly not even a decidable one, since it involves testing whether a rule "could apply to some tree in that derivation in two different ways, such that some variable in the rule instantiated to two different nodes in the tree *of different depth*" (emphasis in original), and given the Turing-equivalent power of the systems Lakoff is trying to mimic in MTS mode, this cannot be decidable, by Rice's theorem (see Hopcroft and Ullman 1979: 185–189.)

Problems similar to those that arise in trying to state the cyclic principle also arise in trying to define rule ordering,

optional versus obligatory application, and other matters (see Soames 1974). The bottom line is that it is wildly off the mark to suggest, as Postal (1972: 139) did, that Lakoff's work provides "a fundamental theoretical clarification".[3]

I am not suggesting that Lakoff's project was inherently impossible to execute. It might have been feasible. One clearly needed improvement was provision of a way to identify *corresponding nodes* in different trees in the sequence directly (see Soames 1974: 127 on this). Lakoff seems to presuppose that corresponding nodes can be located when necessary, but he does not define a correspondence relation that might make it feasible. Potts and Pullum (2002), in the course of applying MTS description to the content of constraints in optimality-theoretic phonology, assume structures that are in effect tree pairs with an added correspondence relation $\mathfrak{R}$ defined between the nodes in the first tree and the nodes in the second. Lakoff could have taken structures to be single connected graphs — tree sequences with the nodes linked to corresponding nodes in successor trees by the $\mathfrak{R}$ relation.

An analog of the highly complex *ceteris paribus* condition would still be needed, representing an embarrassingly massive redundancy in the structural representations involved. And it still needs to be shown that TG could be recast with identical descriptive power in terms of sets of conditions on graphs of the relevant sort. Lakoff cannot be taken to have done anything more than adumbrate the approach. As Zwicky (1972: 106) remarks, it is unfortunate that Lakoff and others who read it "responded to the tone of Chomsky's article rather than to its actual content."

9. More recent work   The failure of Lakoff's project might look like a very serious strike against the idea of a grammar as a set of constraints if the models for constraints in natural language syntax had to be of the sort Lakoff assumes. But of course they do not.

Lakoff's reasons for assuming models having the form of transformational derivations (tree sequences with some way of determining a counterpart relation for most of the nodes between successive trees) appear to have been bound up with his effort to show, within the analytical framework of TG, that the assumptions of Chomsky (1965) led inexorably toward the generative semantics hypothesis. Lakoff had a primarily rhetorical motive, in other words: he wanted to reduce to meaninglessness the question of whether deep structures or semantic representations are 'generated first'. He was not concerned with the question of whether simpler models of the structure

of expressions might suffice for adequate descriptions of natural languages. That possibility was to be explored in later research.

Subsequent work by Freidin (1975), Brame (1976), and Bresnan (1978) led to a widespread weakening of the idea that transformational description was inevitable; and Gazdar (1981) finally made it explicit that perhaps purely context-free description had been much underestimated.

The 1980s saw a diverse array of developments in syntactic theory (see Jacobson and Pullum (1982) for a snapshot of the field at the beginning of the decade), but the most relevant in the present context was the arc pair grammar (APG) framework of Johnson and Postal (1980). This was the first moderately complete proposal for an MTS syntactic framework. It emerged from the relational grammar tradition, but was hardly in contact with the rest of linguistics at all.

In APG, a structure is a triple $\mathcal{A} = \langle A, R_s, R_e \rangle$, where A is a set of arcs (roughly, an arc is an edge labeled with a grammatical relation like 'subject-of' and a sequence of stratum indices) and R_s and R_e are binary relations. R_s is called **sponsor**: intuitively, $R_s(A_1, A_2)$ means that the presence of A_1 is a necessary condition for A_2 to be in the structure. R_e is called **erase**: the intuition is that $R_e(A_1, A_2)$ means the presence of A_1 is sufficient condition for A_2 to have no relevance to any superficial properties like word order, morphology, and phonology.[4]

Johnson and Postal state, in what appears to be a first-order language enriched with the power to define reflexive transitive closures (they do not specify their description language with care), a large number of proposed universal laws of syntax and a number of proposed rules for English and other languages, but they also draw (chap. 14) a number of consequences from the idea of MTS theories, such as the observation that rules and universal principles can be stated in exactly the same logical language and have models of exactly the same sort; the point that multiple coordination with branching of unbounded degree becomes easily describable; and the suggestion that syntax can be separated completely from the lexicon, making possible an explanation of the intelligibility of expressions containing nonsense words.

Meanwhile there was a surprising (and temporary) turn taken by TG during the early 1980s, when GB started framing a significant part of the general theory of syntax in declarative terms ('An anaphor is bound in its governing category', and so on). However, there was no attempt by the practitioners of such theories to formalize them,

---

[3]Postal apparently said this because he agreed with Lakoff that rule ordering in a GES grammar should be regarded as just another descriptive device like positing an extra rule. This seems sensible. But since Lakoff had no workable way of representing ordering of transformations, he can hardly be credited with having provided a theoretical clarification of them.

[4]Johnson and Postal do not formalize things quite this way. They treat nodes as primitive, and define an arc as a pair of nodes associated with a grammatical relation name and a sequence of stratum indices, and then with some awkwardness treat **sponsor** and **erase** as higher-order relations between arcs. It seems preferable to formalize the theory in terms of edges as primitives, as Postal has suggested in unpublished work.

and while the binding theory and case theory seemed implicity model-theoretic in conception, X-bar theory and Move Alpha were clearly GES ideas. GB was an rather informally developed hybrid framework: a little casual declarative superstructure built on top of an underlyingly procedural core. The conceptual nod toward the idea of giving theories in a form that involves statements about structures can be acknowledged, but it would be too great a stretch to call GB a part of the MTS project.

Elsewhere during the later 1980s there were only occasional hints of the MTS perspective, often in unpublished or fairly obscure work: the lectures Gerald Gazdar gave in 1987 advocating a fully satisfaction-based formalization of GPSG; the ideas Ron Kaplan expressed in the late 1980s concerning LFG as using a quantifier-free equational logic on complex models incorporating functions (see Kaplan (1995), which dates from 1989); the far-sighted work by Paul John King (1989) on development of an MTS formalization of HPSG; and so on.

Basically, MTS as a full-fledged variety of linguistic theorizing can be said to have begun with Johnson and Postal (1980). So there is a sense in which MTS is not just 10 years old this year, but more like 30. But it is of course artificial to give precise ages to intellectual movements. Like words and phrases in the history of a language, they always turn out to be a little older than the last investigator thought. What is certainly clear is that the MTS project mostly languished between 1980 and about 1993. Hardly anybody paid attention to arc pair grammar, and the one or two who did (e.g., Judith Aissen) were interested in its hypotheses about syntactic structure and its inventory of conjectured syntactic universals (see Aissen 1987 for an essentially unique APG-based descriptive study).

It was only in the 1990s, as computational linguists with a training in logic became involved, that MTS work with some real mathematical and logical sophistication began to emerge. A partial timeline:

1993: Kracht (1993) (partly inspired by Barker and Pullum 1990) and Blackburn et al. (1993) (re-formalizing Gazdar et al. 1985), both from German institutions and using modal logic on tree models, presented at the 6th EACL meeting in Dublin.

1994: James Rogers completes a dissertation at the University of Delaware (Rogers, 1994) using wMSO on tree models; Blackburn, Kracht, and Rogers meet at a workshop in Amsterdam ('Logic, Structures and Syntax, at the Centrum voor Wiskunde en Informatica, September 26–28).

1995: Backofen et al. (1995) publish their first-order axiomatization of the theory of finite trees; Kracht publishes two relevant papers (Kracht 1995a; Kracht 1995b; Kaplan (1995) publishes a clearly MTS-oriented statement of the bases of lexical functional grammar, and (Blackburn and Gardent, 1995) publish a different reformalization using modal logic.

1996: Rogers presents a paper at the first conference on Logical Aspects of Computational Linguistics (LACL); ESSLLI (in Prague; see `http://folli.loria.fr/esslliyear.php?1996`) features an advanced course by Rogers called 'Topics in Model-Theoretic Syntax' — a title that Rogers proposed as a joke but was persuaded by Blackburn to keep — and also a workshop organized by Uwe Mönnich and Hans Peter Kolb of Tübingen under the title 'The Mathematics of Syntactic Structure'.

1997: proceedings of the 1994 Amsterdam workshop appear as Blackburn and de Rijke (1997); Rogers' term 'model-theoretic syntax' appears in print for the first time in the title of Blackburn and Meyer-Viol (1997); Rogers' LACL paper published as Rogers (1997b); Rogers (1997a) uses MTS to reformalize aspects of GPSG.

The explosion of MTS publication in 1997 makes it very appropriate to be holding a tenth-anniversary reunion in 2007. I have tried to point out in the brief historical review above, however, is that the flowering of this work that began in the middle 1990s was related to seeds planted some thirty years before. They were planted in stony ground, only inexpertly tended, and inadequately watered, but they were planted nonetheless. There is now an increasingly luxuriant garden to explore.

References

Aissen, Judith (1987). *Tzotzil Clause Structure*. Kluwer Academic, Dordrecht.

Backofen, Rolf, James Rogers, and K. Vijay-shanker (1995). A first-order axiomatization of the theory of finite trees. *Journal of Logic, Language, and Information*, **4**:5–39.

Barker, Chris and Geoffrey K. Pullum (1990). A theory of command relations. *Linguistics and Philosophy*, **13**:1–34.

Blackburn, Patrick and Maarten de Rijke, eds. (1997). *Specifying Syntactic Structures*. Studies in Logic, Language and Information. CSLI Publications and FoLLI, Stanford, CA.

Blackburn, Patrick and Claire Gardent (1995). A specification language for lexical functional grammars. In *Seventh Conference of the European Chapter of the Association for Computational Linguistics*, pp. 39–44. ACL, Morristown, NJ.

Blackburn, Patrick, Claire Gardent, and Wilfried Meyer-Viol (1993). Talking about trees. In *Sixth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 21–29. ACL, Morristown, NJ.

Blackburn, Patrick and Wilfried Meyer-Viol (1997). Modal logic and model-theoretic syntax. In M. de Ri-

jke, ed., *Advances in Intensional Logic*, pp. 29–60. Kluwer Academic, Dordrecht.

Brame, Michael K. (1976). *Conjectures and Refutations in Syntax and Semantics*. Elsevier North-Holland, New York.

Bresnan, Joan W. (1978). A realistic transformational grammar. In *Linguistic Theory and Psychological Reality*, pp. 1–59. MIT Press, Cambridge, MA.

Chomsky, Noam (1956). The logical structure of linguistic theory. Unpublished dittograph, microfilmed; revised version of a 1955 unpublished manuscript.

Chomsky, Noam (1959). On certain formal properties of grammars. *Information and Control*, **2**:137–167.

Chomsky, Noam (1962). Explanatory models in linguistics. In Ernest Nagel, Patrick Suppes, and Alfred Tarski, eds., *Logic, Methodology and Philosophy of Science: Proceedings of the 1960 International Congress*, pp. 528–550. Stanford University Press, Stanford, CA.

Chomsky, Noam (1965). *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.

Chomsky, Noam (1972). Some empirical issues in the theory of transformational grammar. In P. Stanley Peters, ed., *Goals of Linguistic Theory*. Holt Rinehart and Winston, New York.

Chomsky, Noam (1975). *The Logical Structure of Linguistic Theory*. Plenum, New York. Published version of Chomsky (1956).

Doner, John (1970). Tree acceptors and some of their applications. *Journal of Computer and System Sciences*, **4**:406–451.

Freidin, Robert (1975). The analysis of passives. *Language*, **51**:384–405.

Gazdar, Gerald (1981). Unbounded dependencies and coordinate structure. *Linguistic Inquiry*, **12**:155–184.

Gazdar, Gerald (1982). Phrase structure grammar. In Pauline Jacobson and Geoffrey K. Pullum, eds., *The Nature of Syntactic Representation*, pp. 131–186. D. Reidel, Dordrecht, Netherlands.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum, and Ivan A. Sag (1985). *Generalized Phrase Structure Grammar*. Basil Blackwell, Oxford.

Gazdar, Gerald and Geoffrey K. Pullum (1981). Subcategorization, constituent order, and the notion 'head'. In Michael Moortgat, Harry van der Hulst, and Teun Hoekstra, eds., *The Scope of Lexical Rules*, pp. 107–123. Foris, Dordrecht, Netherlands.

Hopcroft, John E. and Jeffrey D. Ullman (1979). *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley.

Jacobson, Pauline and Geoffrey K. Pullum, eds. (1982). *The Nature of Syntactic Representation*. D. Reidel, Dordrecht. Republished by Springer Verlag.

Johnson, David E. and Paul M. Postal (1980). *Arc Pair Grammar*. Princeton University Press, Princeton, NJ.

Kaplan, Ronald (1995). The formal architecture of lexical-functional grammar. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell III, and Annie Zaenen, eds., *Formal Issues in Lexical-Functional Grammar*, pp. 7–27. CSLI Publications, Stanford, CA. Earlier versions appeared in *Proceedings of ROCLING II*, ed. by C.-R. Huang and K.-J. Chen (Taipei, Republic of China, 1989), 1–18, and in *Journal of Information Science and Engineering* **5**, 1989, 305–322.

King, Paul John (1989). *A Logical Formalism for Head-driven Phrase Structure Grammar*. Ph.D. thesis, University of Manchester, Manchester, UK.

Kornai, Andràs and Geoffrey K. Pullum (1990). The X-bar theory of phrase structure. *Language*, **66**:24–50.

Kracht, Marcus (1993). Mathematical aspects of command relations. In *Sixth Conference of the European Chapter of the Association for Computational Linguistics: Proceedings of the Conference*, pp. 240–249. Association for Computational Linguistics, Morristown, NJ.

Kracht, Marcus (1995a). Is there a genuine modal perspective on feature structures? *Linguistics and Philosophy*, **18**:401–445.

Kracht, Marcus (1995b). Syntactic codes and grammar refinement. *Journal of Logic, Language and Information*, **4**:41–60.

Lakoff, George (1969). On derivational constraints. In Robert I. Binnick, Alice Davison, Georgia Green, and Jerry Morgan, eds., *Papers from the Fifth Regional Meeting, Chicago Linguistic Society*, pp. 117–139. Department of Linguistics, University of Chicago, Chicago, Illinois.

Lakoff, George (1971). On generative semantics. In Danny D. Steinberg and Leon A. Jakobovits, eds., *Semantics: An Interdisciplinary Reader in Philosophy, Linguistics and Psychology*, pp. 232–296. Cambridge University Press, Cambridge.

Lasnik, Howard (2000). Syntactic Structures *Revisited: Contemporary Lectures on Classic Transformational Theory*. MIT Press, Cambridge, MA.

McCawley, James D. (1968). Concerning the base component of a transformational grammar. *Foundations of Language*, **4**:243–269. Reprinted in James D. McCawley, *Grammar and Meaning*, 35–58 (New York: Academic Press; Tokyo: Taishukan, 1973).

Post, Emil (1943). Formal reductions of the general combinatory decision problem. *American Journal of Mathematics*, **65**:197–215.

Postal, Paul M. (1972). The best theory. In P. Stanley Peters, ed., *Goals of Linguistic Theory*, pp. 131–179. Prentice-Hall, Englewood Cliffs, NJ.

Potts, Christopher and Geoffrey K. Pullum (2002). Model theory and the content of OT constraints. *Phonology*,

19:361–393.

Pullum, Geoffrey K., James Rogers, and Barbara C. Scholz (in preparation). *Model-Theoretic Syntax*. Oxford University Press, Oxford.

Pullum, Geoffrey K. and Barbara C. Scholz (2001). On the distinction between model-theoretic and generative-enumerative syntactic frameworks. In Philippe de Groote, Glyn Morrill, and Christian Retoré, eds., *Logical Aspects of Computational Linguistics: 4th International Conference*, number 2099 in Lecture Notes in Artificial Intelligence, pp. 17–43. Springer, Berlin and New York.

Pullum, Geoffrey K. and Barbara C. Scholz (2005). Contrasting applications of logic in natural language syntactic description. In Petr Hajek, Luis Valdes-Villanueva, and Dag Westerstahl, eds., *Proceedings of the 13th International Congress of Logic, Methodology and Philosophy of Science*, pp. 481–503. KCL Publications, London.

Rogers, James (1994). *Studies in the Logic of Trees with Applications to Grammar Formalisms*. Ph.D. thesis, University of Delaware, Newark, DE.

Rogers, James (1997a). "Grammarless" phrase structure grammar. *Linguistics and Philosophy*, **20**:721–746.

Rogers, James (1997b). Strict LT_2 : regular :: local : recognizable. In Christian Retoré, ed., *Logical Aspects of Computational Linguistics: First International Conference, LACL '96 (Selected Papers)*, number 1328 in Lecture Notes in Artificial Intelligence, pp. 366–385. Springer, Berlin and New York.

Rogers, James (1999). The descriptive complexity of generalized local sets. In Hans-Peter Kolb and Uwe Mönnich, eds., *The Mathematics of Syntactic Structure: Trees and their Logics*, number 44 in Studies in Generative Grammar, pp. 21–40. Mouton de Gruyter, Berlin.

Scholz, Barbara C. and Geoffrey K. Pullum (2007). Tracking the origins of transformational generative grammar. *Journal of Linguistics*, **43**(3). In press.

Soames, Scott (1974). Rule orderings, obligatory transformations, and derivational constraints. *Theoretical Linguistics*, **1**:116–138.

Soare, Scott (1996). Computability and recursion. *Bulletin of Symbolic Logic*, **2**:284–321.

Thompson, Henry (1975). The cycle: a formal statement. In Robin E. Grossman, L. James San, and Timothy J. Vance, eds., *Papers from the Eleventh Regional Meeting, Chicago Linguistic Society*, pp. 589–603. Chicago Linguistic Society, Chicago, Illinois.

Zwicky, Arnold M. (1972). Remarks on directionality. *Journal of Linguistics*, **8**:103–109.

Zwicky, A. M. and S. Isard (1963). Some aspects of tree theory. Working Paper W-6674, The MITRE Corporation, Bedford, MA.