Syntactic Structures and Recursive Devices: The Perils of Imprecision

Marcus Tomalin

1 50 Years Ago...

Syntactic Structures (*SS*) was published in 1957 (most likely in February; the Preface is dated August 1st 1956) as part of the *Janua Linguarum* series that was edited by Cornelis van Schooneveld. Later in the same year, Robert Lees published an influential review of *SS* in the journal *Language* (July-September, Vol. 33).

SS is 116 page text with the following contents:

- Preface
- 1. Introduction
- 2. The Independence of Grammar
- 3. An Elementary Linguistic Theory
- 4. Phrase Structure
- 5. Limitations of Phrase Structure Description
- 6. On the Goals of Linguistic Theory
- 7. Some Transformations in English
- 8. The Explanatory Power of Linguistic Theory
- 9. Syntax and Semantics
- Summary
- Appendix I: Notations and Terminology
- Appendix II: Examples of English Phrase Structure and Transformational Rules
- Bibliography

The following quotations indicate a range of contrasting perspectives concerning *SS*:

> [...] Chomsky's book on syntactic structures is one of the first serious attempts on the part of a linguist to construct within the tradition of scientific theory-construction a comprehensive theory of language which may be understood in the same sense that a chemical, biological theory is ordinarily understood by experts in those fields. It is not a mere reorganisation of the data into a new kind of library catalogue, nor another speculative philosophy about the nature of Man and Language, but rather a rigorous explication of our intuitions about our language in terms of an overt axiom system [...] (Lees 1957: 377-378)

> [*Syntactic Structures*] completely shattered the prevailing structuralist conceptions of linguistic theory. (Newmeyer 1980: 35)

> At the time Mouton was publishing just about anything, so they decided they'd publish it along with a thousand other worthless things that were coming out. That's the story of *Syntactic Structures*: course notes for undergraduate science students published by accident in Europe. (Chomsky, quoted in Dillinger and Palácio 1997: 162-163)

SS is standardly seen as having been incomparably radical and revolutionary, though recent studies have questioned this well-established belief (for instance, see Matthews 1993 and Tomalin 2006).

Indeed, *SS* is an odd text. It is (in effect) a high-level summary of work that Chomsky had previously presented elsewhere, and the main sources are:

- *Morphophonemics of Modern Hebrew* (1951)
- 'Systems of Syntactic Analysis' (1953)
- 'Logical Syntax and Semantics: their linguistic relevance' (1955)
- 'Semantic Considerations in Grammar' (1955)
- *Logical Structure of Linguistic Theory* (1955)
- 'Three Models for the Description of Language' (1956)

Despite that fact that Lees (and others) considered *SS* to be 'a rigorous explication', Chomsky frequently apologises for the non-rigorous, high-level, and incomplete nature of the text:

- '[...] this fact may be obscured by the informality of the presentation' (Chomsky 1957: 5)
- 'See my 'Three models for the description of language' [...] for proofs about this and related theorems about relative power of grammars' (Chomsky 1957: 30)
- '[...] this is a complex matter that requires a much more detailed development of transformational theory that we can give it here' (Chomsky 1957: 77)

The informality manifests itself elsewhere too. *SS* is a fairly short monograph, yet there are many typos/errors that suggest poor/rushed proof-reading and copy editing:

- '[...] from the viewpoint of transformational nalysis' (Chomsky 1957: 68)
- 'the boy studying in the ljbrary' (Chomsky 1957: 81)
- 'SYNTACS AND SEMANTICS' (Chomsky 1957: 93, 99; running chapter heading)
- 'T^{op}_{sub}' (Chomsky 1957: 81) – should be T^{ob}_{sub}!

So, *SS* was an informal (error-laden), high-level summary of existing research, aimed at an undergraduate audience, and yet it was considered by professional linguists to be so 'rigorous' and so revolutionary that it 'shattered' prevailing notions in linguistic theory.

This talk will attempt to probe the intriguing character of *SS* by focusing on a particular topic – namely recursion. The basic intentions are as follows:

- to reflect upon the role of recursion within *SS* (and some of Chomsky's other publications from the 1950s) in order to demonstrate both the strengths and weaknesses of *SS*.
- to trace the origins of Chomsky's understanding of recursion
- to relate the discussions in the 1950s literature to more recent debates about the nature of recursion.

2 What is Recursion?

Before focusing on *SS*, a brief overview of 'recursion' may be helpful:

- **OED**: 'The application or use of a recursive procedure or definition; primitive recursion, definition of a function of natural numbers by induction on a single argument or (equivalently) by simple recursion formulae; recursion formula, an equation relating the value of a function for a given value of its argument (or arguments) to its values for other values of the argument(s)'.

- **Wikipedia**: 'Recursion, in mathematics and computer science, is a method of defining functions in which the function being defined is applied within its own definition. The term is also used more generally to describe a process of repeating objects in a self-similar way.' (July 2007)

Certain keywords appear frequently when recursion is discussed:

- recursive function, recursive definition, recursive device, recursive procedure
- self-reference, circularity
- infinity
- repetition, iteration
- efficiency

Does the following definition provide an answer to the question posed in this section?

> *Recursion*:
> If you still don't know, See: "Recursion".

3 Chomsky and Recursion

3.1 Recursion in *Syntactic Structures*

Recursion is first mentioned in Chapter 3 of *SS* during a discussion of Finite State Machines (FSMs). Chomsky notes that he had encountered FSMs in Shannon and Weaver 1949 (though they don't discuss recursion in their presentation). A few extracts:

- FSM = '[a] familiar communication theoretic model for language' (Chomsky 1957: 18)
- A given FSM has 'a finite number of different internal states', including 'an *initial state*' and 'a *final state*' (Chomsky 1957: 18-19)
- 'we assign a probability to each transition from state to state' (Chomsky 1957: 20)
- 'Any language that can be produced by a machine of this sort we call a *finite state language*; and we can call the machine itself a *finite state grammar*' (Chomsky 1957: 19)
- Finite State Grammars (FSGs) can contain 'closed loops', and these loops enable the grammar 'to produce a infinite number of sentences' (Chomsky 1957: 19)

As an example of a simple FSG with a closed loop, Chomsky presents the following diagram (Chomsky 1957: 19).

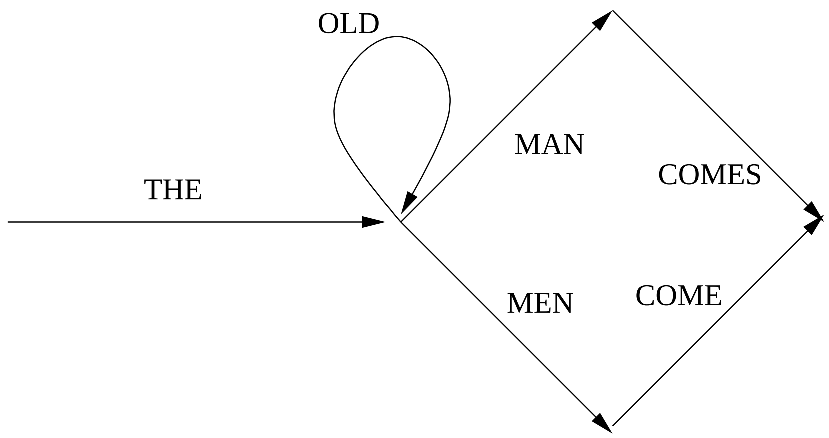

Figure 1: FSG with a closed loop

This FSG enables an infinite set of sentences to be generated: 'the old man comes', 'the old old man comes', 'the old ... old man comes', 'the old men come', 'the old old men come', '' the old ... old men come'.

Later Chomsky notes that:

> In general, the assumption that languages are infinite is made in order to simplify the description of these languages. If a grammar does not have recursive devices (closed loops, as in [Fig.1], in the finite state grammar) it will be prohibitively complex. If it does have recursive devices of some sort, it will produce infinitely many sentences. (Chomsky 1957: 23-24)

This short passage is troublesome:

- it is odd to claim that the infinite generativity of natural language is assumed solely in order to simplify the analysis of natural language!
- no *definitions* of 'recursion' or 'recursive devices' are presented (the closed loop is simply offered as an *example* of a 'recursive device' in the FSG framework)
- 'recursive devices' are presented as being useful (though not *essential*): they enable an infinite number of sentences to be generated without prohibitive complexity in the grammar

At the start of Chapter 4 (in a long footnote), Chomsky briefly considers a 'system of word class analysis' that had been proposed by F. Harwood in 1955, noting that Harwood's system is 'similar' to a phrase structure grammar. He rejects Harwood's approach, though, partly because 'it could not generate an infinite language with a finite grammar' (Chomsky 1957: 27). Since 'recursive devices' enable 'infinitely many sentences' to be produced, seemingly Harwood's system is inadequate since it does not explicitly contain any such devices.

So, *SS* indicates that 'recursion' is of considerable importance when grammars are constructed, but what does the term 'recursion' actually indicate? *SS* offers no definitions, but Chomsky's earlier publications can be explored...

3.2 Recursion in the Pre-*Syntactic Structures* Publications

The extract from *SS* quoted above appears almost verbatim in Chomsky's 1956 paper 'Three Models for the Description of Language' ('TMDL'):

> In general, the assumption that languages are infinite is made for the purpose of simplifying the description. If a grammar has no recursive steps (closed loops, in the model discussed above) it will be prohibitively complex – it will, in fact, turn out to be little better than a list of strings or of morpheme class sequences in the case of natural language. If it does have recursive devices, it will produce infinitely many sentences. (Chomsky 1956: 115-116)

'TMDL' offers a more detailed technical/mathematical account of finite state, phrase structure, and transformational grammars. Chomsky lists *SS* as 'to appear' in his references for 'TMDL' – so which text was finished first? It is most likely that the more detailed account ('TMDL') was summarised in *SS*. However, there is no discussion of closed loops in 'TMDL', and no further discussion of recursion.

Throughout *SS*, Chomsky states repeatedly (often in footnotes) that many of the ideas he is considering were presented in greater detail in his mimeographed *The Logical Structure of Linguistic Theory* (1955; published 1975; from henceforth *LSLT*). In *LSLT* recursive devices are used in various ways...

Having specified a set of n conversions or 're-write' rules (e.g., X $\rightarrow$ Y) in the Phrase Structure component of the grammar, Chomsky notes that certain rules can be recursive:

> When we turn to the level of phrase structure, we find that certain rules may have a recursive character. Thus *Noun Phrase* (NP) might be analyzed in such a way that one of its components may be a NP as in such sentences as "the man who made the discovery is my brother" [...] (Chomsky 1975[1955]: 171-172)

Here ‘recursive’ is associated with syntactic embedding – that is, a phrase (NP_2) is embedded within a phrase of the same type (NP_1) creating structures such as [$_{NP_1}$ the man who made [$_{NP_2}$ the discovery]], and conversions that take the form $NP_1 \rightarrow ...NP_2...$ create structures of this kind.

However, Chomsky later claims in *LSLT* that conversions themselves (whether recursive or not) can be *applied* recursively too:

> [...] we can understand the linear grammar to be the sequence of conversions $S_1, ..., S_n$, $S_1, ..., S_n$, $S_1, ..., S_n, ...$ We then say that a derivation D is *recursively produced* by the linear grammar $S_1, ..., S_n$. We define a *proper linear grammar* as a linear grammar which is so constructed that it is impossible to run through it over and over again vacuously. (Chomsky 1975[1955]: 194-195)

However, the repeated application of a set of rules is not necessarily associated with the idea of embedded syntactic structures. Towards the end of *LSLT*, Chomsky considered whether ‘recursive parts of the grammar’ should be permitted in the phrase structure component, or whether they should only appear in the transformational component (Chomsky 1975[1955]: 516-518).

So, *LSLT* offers various examples of recursion in linguistic theory, but (as in *SS* and ‘TMDL’) no definitions are presented, and therefore it is assumed that the term in unproblematical. Other 1950s publications, though, indicate some of the sources of Chomsky’s thinking about these topics. In his paper ‘Logical Syntax and Semantics: their linguistic relevance’ he refers to the work of Yehoshua Bar-Hillel, and notes that

> [a]t one point, Bar-Hillel suggests that recursive definitions may be useful in linguistic theory; whether this turns out to be the case or not, I agree in this instance with the spirit of his remarks. (Chomsky 1955: 45)

This suggests that Chomsky had encountered recursive definitions in Bar-Hillel’s work while he was working on *LSLT*. Consequently, the origins of his understanding of ‘recursion’ must be traced even further back...

4 A Little History...

4.1 Inductive Definitions

In 1889, Giuseppe Peano published his axioms for the natural numbers. In Peano’s system, ‘$\mathbf{N}$’ means ‘positive integer’, ‘1’ means ‘unity’, ‘$a + 1$’ means ‘the successor of a’, ‘$x \in y$’ means ‘x is in y’, while ‘$x \supset y$’ means ‘x implies y’, and, given these ‘explicationes’ (Peano 1958[1889]: 34), Peano states his 1st and 6th axioms as follows:

$$1 \in \mathbf{N} \tag{1}$$

$$(a \in \mathbf{N}) \supset (a + 1 \in \mathbf{N}) \tag{2}$$

Axiom (1) states that ‘1’ is a natural number, while axiom (2) states that the successor of a natural number is also a natural number, and, taken together, these axioms provide an inductive definition for the set of natural numbers: given (1), the repeated application of (2) enables the set of natural numbers to be obtained, even though this set is potentially infinite.

4.2 Recursive Function Theory

Kurt Gödel developed his theory of recursive functions in the late 1920s and early 1930s, and this theory, which included his definition of primitive recursive functions, became a focus of general research in 1931 when he used these functions in his celebrated incompleteness theorem. In his 1931 paper, Gödel defined primitive recursive functions as follows:

$$\begin{aligned} \phi(0, x_2, ..., x_n) &= \psi(x_2, ..., x_n) \\ \phi(k+1, x_2, ..., x_n) &= \mu(k, \phi(k, x_2, ..., x_n), x_2, ..., x_n) \end{aligned} \tag{3}$$

In (3) the number-theoretic function $\phi(x_1, x_2, ..., x_n)$ is recursively defined in terms of the number-theoretic functions $\psi(x_1, x_2, ...x_{n-1})$ and $\mu(x_1, x_2, ..., x_{n+1})$, assuming that these hold for $x_1, ..., x_{n+1}$, and k.

Gödel soon realised that there were certain functions which could not be classified as *primitive* recursive, but which were nonetheless recursive. For instance, if $\psi(y)$ and $\chi(x)$ are primitive recursive, then the function $\phi(x, y)$ can be defined inductively by the following relations:

$$\begin{aligned} \phi(0, y) &= \psi(y) \\ \phi(x+1, 0) &= \chi(x) \\ \phi(x+1, y+1) &= \phi(x, \phi(x+1, y)) \end{aligned} \tag{4}$$

In this case, the function ϕ is defined (by induction) with respect to two variables simultaneously. Consequently, it is not primitive recursive. However, in order to enable such functions to be classified as recursive, Gödel introduced the notion of 'general recursive functions' (see Gödel 1934: 368-369).

4.3 Recursion, λ-definability, and Turing-computable Functions

By 1934, Gödel had provided clear definitions of primitive and general recursive functions. However, during the 1930s and 1940s various developments complicated the situation:

- λ-definability was introduced in 1936 by Alonzo Church, and this introduced the notion of 'effectively calculable' (Church 1936)
- Church demonstrated that '[e]very [general] recursive function of positive integers is λ-definable' and '[e]very λ-definable function of positive integers is [general] recursive' (Church 1936: 349)
- computability theory was introduced by Alan Turing in 1936, and this introduced the notion of (Turing-)computable functions (Turing 1936)
- Turing proved that 'all effectively calculable (λ-definable) sequences are computable', and vice versa (Turing 1936: 263)

Consequently, by the end of the 1930s it was established that three distinct mathematical frameworks – (general) recursive functions, λ-definable functions, and Turing-computable functions – were equivalent.

4.4 Recursively Enumerable Sets

In unpublished research from the 1920s, Emil Post had largely anticipated the results of Gödel, Church, and Turing. During the 1940s, he elaborated the notion of recursively enumerable sets – especially in his 1944 paper 'Recursively Enumerable Sets of Positive Integers and Their Decision Problems'. [NB: like *SS*, Post 1944 is an informal/semi-formal presentation of a formal subject...]

Post defined recursively enumerable sets as follows:

> A set of positive integers is said to be *recursively enumerable* if there is a recursive function $f(x)$ of one positive integral variable whose sequence $f(1), f(2), f(3), ...$ is then said to be a *recursive enumeration* of the set. (Post 1944: 285)

Further, if a set, S, is recursively enumerable, and if its complement, $\overline{S}$, is recursively enumerable, then S is said to be 'recursive' and therefore 'there is an effective method for telling of any positive integer n whether it is, or is not, in S' (Post 1944: 290). This distinction between 'recursively enumerable' and 'recursive' sets enabled him to established theorems such as the following:

> **Theorem:** There exists a recursively enumerable set of positive integers which is not recursive. (Post 1944: 291)

Consequently (and crucially), Post used 'recursive' to refer both to certain mathematics objects (e.g., sets), and to certain devices/techniques (e.g., functions) that were used to generate them.

Importantly, the results obtained by Peano, Gödel, Church, Turing, Post, and others were influentially summarised in Stephen Kleene's textbook *Introduction to Metamathematics* (1952), and this is a text with which many mathematically-minded linguistics in the 1950s were familiar.

5 From Mathematics to Linguistics

5.1 Bar-Hillel and Recursion

And so back to Bar-Hillel. In his 1953 paper 'On Recursive Definitions in Empirical Science', he argued that recursive definitions should be used in the empirical sciences, such biology and linguistics, as well as formal sciences, such as mathematics and logic.

Modifying an example that Kleene had discussed (see Kleene 1952: 218), Bar-Hillel gives the following example of a recursive definition from the formal sciences:

$$
\begin{aligned}
a + 1 &= a' && (5)\\
a + n' &= (a + n)' && (6)
\end{aligned}
$$

observing that, although this pair of equations may look 'flagrantly circular' (Bar-Hillel 1953: 160), it is not, since the '+' can be eliminated:

$$
\begin{aligned}
4 + 3 &= 4 + 2' && (7)\\
&= (4 + 2)' && (8)\\
&= (4 + 1')' && (9)\\
&= (4 + 1)'' && (10)\\
&= 4''' && (11)\\
& && (12)
\end{aligned}
$$

He concludes:

> In our case, the attempt at elimination did not result in an infinite regress. The dreaded circle turned out not to be a vicious one. Strictly speaking, it is not a circle at all but a harmless terminating spiral. (Bar-Hillel 1953: 161)

He then turns from mathematics to linguistics. Using English as a 'metalanguage' and French as 'an object language', and he introduces the following definition (see Bar-Hillel 1953: 163):

Definition 5.1: Sentence (Recursive in Disguise)
x will be called a *sentence* (in French) if (and only if) x is a sequence of a nominal and a (intransitive) verbal, or a sequence of a nominal, a (transitive) verbal and a nominal or, ..., or a sequence of a sentence, the word "et", and a sentence, or, ...

He argues that, although Def.5.1 does not have the two-part structure of a recursive definition, it is 'recursive in disguise' and can be re-written as follows (see Bar-Hillel 1953: 163):

Definition 5.2: Sentence (Recursive)
1. x is a *sentence*$_1$ (a *simple sentence*) $=_{\mathrm{df}}$ x is a sequence of a nominal and a (intransitive) verbal or a sequence of a nominal, a (transitive) verbal, and a nominal, or ...

2. x is a *sentence*$_{n+1}$ (a *compound sentence* of the $n+1^{th}$ order) $=_{\mathrm{df}}$ x is a sequence of a sentence$_p$, the word "et", and a sentence$_m$, where either p or m (or both) are equal to n and none is greater than n, or ...

He notes that Def.5.2 clearly takes the form of 'a pair of simultaneous recursive definitions', and he claims that it is 'simple' to check whether a given compound sentence is 'a proper French sentence' or not.

In a subsequent paper, 'Logical Syntax and Semantics' (1954), Bar-Hillel suggested that recursive definitions could be used specifically in order to clarify the nature of the relationship between the various postulated levels of linguistic analysis (e.g., phonological, morphological). He developed his ideas responding to the work of Kenneth Pike (and others).

Pike had argued that these levels should not be entirely separate, and he outlined a 9-step procedure that would enable a linguist to produce a phonetic transcription of a spoken utterance (see Pike 1952). Bar-Hillel claimed that

- certain linguists wanted to keep the levels separate so that the whole system would not 'in constant danger of succumbing to an infestation of meaning' (Bar-Hillel 1954: 234).
- linguists were worried about the circularity of defining phonemes in terms of morphemes while simultaneously defining the latter in terms of the former

It was possible, though, to use seemingly circular 'concept formations' that avoid infinite regress:

> [...] concept formations of these kinds are in regular use in mathematics, and especially in mathematical logic, where they are known as a specifical case of RECURSIVE DEFINITIONS. It seems rather likely (though a detailed proof would require many man-hours of work) that Pike's nine-step procedure can be formalized and adequately represented by a set of such definitions. (Bar-Hillel 1954: 234)

As indicated earlier, Chomsky responded positively to this suggestion in 1955. However, by the time he wrote *SS*, he seems to have changed his mind:

> Bar-Hillel has suggested in "Logical syntax and semantics", *Language* 30.230-7 (1954) that Pike's proposals can be formalized without the circularity that many sense in them by the use of recursive definitions. He does not pursue this suggestion in any detail, and my own feeling is that success along these lines is unlikely. (Chomsky 1957: 57-58)

So, *SS* suggests that recursive definitions and/or 'devices' are sometimes helpful (e.g., closed loops), and sometimes unhelpful (e.g., when defining linguistic levels)...

5.2 Recursion and Ambiguity

A decade ago, the mathematician Robert Soare referred to the ambiguity that surrounds the term 'recursion' as the 'Recursion Convention', and he highlighted the dangers of this imprecision:

> The Recursion Convention has brought "recursive" to have at least four different meanings [...] This leads to some ambiguity [...] Worse still, the Convention leads to *imprecise thinking* about the basic concepts of the subject; the term "recursion" is often used when the term "computability" is meant (By the term "recursive function" does the writer mean "inductively defined function" or "computable function"?) Furthermore, ambiguous and little recognized terms and imprecise thinking lead to *poor communication* both within the subject and to outsiders, which leads to isolation and a lack of progress within the subject, since progress in science depends upon the collaboration of many minds. (Soare 1996: 29)

This passage emphasises current uncertainty – yet the ambiguities that Soare identifies were already prevalent in the 1950s, when Chomsky was writing *SS*. When *SS* appeared, the term 'recursion' and its cognate forms could be interpreted in several distinct (though sometimes related) ways:

- I1: 'recursion' = inductive definition (à la Peano 1889)
- I2: 'recursion' = primitive recursion (à la Gödel 1931, Bar-Hillel 1953 [?])
- I3: 'recursion' = general recursion (à la Gödel 1934)
- I4: 'recursion' = λ-definability (à la Church 1936)
- I5: 'recursion' = computability (à la Turing 1936)
- I6: 'recursion' = generated set (à la Post 1944)
- I7: 'recursion' = 'embedding' conversion (à la Chomsky 1955)
- I8: 'recursion' = repeated application of conversions (à la Chomsky 1955)
- I9: 'recursion' = closed loop (à la Chomsky 1957)

Since 'recursion' was clearly a problematical term in the 1950s, it is intriguing that Chomsky did not attempt to clarify his use of this term in *SS*. He suggests that (some) recursive devices are essential to linguistic theory, but he does not offer any kind of definition, and therefore his discussion is profoundly ambiguous.

This is just one example of the way in which *SS* draws upon the literature and terminology of logic and mathematics, but does not manage to present a stable 'formalized' analytical linguistic theory. *SS* simply inherits the contemporaneous confusion concerning the term 'recursion', and it does not try to clarify.

However, surely these problems are simply due to the fact that, in the 1950s, such research was new and pioneering? Surely everything has been sorted out by now...?

6 The Importance of Recursion?

6.1 Recursion and the Faculty of Language (in the Narrow Sense)

In recent years, recursion has become an increasingly prominent topic (again). In 2002, Hauser, Chomsky, and Fitch defined the Faculty of Language in the Narrow Sense (FLN) as being an abstract linguistic computational system (independent of the other systems with which it interacts and interfaces) which 'takes a finite set of elements and yields a potentially infinite arrangement of discrete expressions' (Hauser, Chomsky, and Fitch 2002: 1571; henceforth HCF).

Given this assumption, HCF claims that

> FLN - the computational mechanism of recursion - is recently evolved and unique to our species [...] we propose in this hypothesis that FLN comprises only the core computational mechanisms of recursion as they appear in narrow syntax and the mappings to the interfaces. (Hauser, Chomsky, and Fitch 2002: 1573)

This implies that FLN (a unique species-specific property) is primarily, and perhaps exclusively, a recursive device (or devices of some kind). However (in a now familiar manner), the informal discussion in HCF offers no definitions – it is assumed that everyone knows/agrees what 'recursion' is.

6.2 Criticisms and Counter-examples

Steven Pinker and Ray Jackendoff have argued that the discussion in HCF is vague. While claiming that phonological structures are not recursive, they observe that:

> [...] the segmental/syllabic aspect of phonological structure, though discretely infinite and hierarchically structured, is not technically recursive. (As mentioned, HCF use "recursion" in a loose sense of concatenation within hierarchically embedded structures). Recursion consists of embedding a constituent in a constituent of the same type, for example a relative clause inside a relative clause (*a book that was written by the novelist you met last night*). This does not exist in phonological structure: a syllable, for instance, cannot be embedded in another syllable. (Pinker and Jackendoff 2005: 10)

In linguistics, then, 'recursion' should indicate syntactic embedding (exclusively), and therefore the HCF argument is misleading.

Dan Everett has rejected the HCF claim that 'recursion' is fundamental to natural language. He argues that the language Pirahã is characterised by 'the absence of embedding' (Everett 2005: 621), where 'embedding' means that one constituent contains a constituent of the same kind:

> English expresses the content of verbs such as "to say", "to think", and "to want" as clausal complements [here the subscript s labels the embedded clauses as theory-neutral]: "I said that [$_s$ John will be here]", "I want [$_s$ you to come]", "I think [$_s$ it's important]". In Pirahã the contents of such verbs, to the degree that equivalent verbs exist at all, are expressed without embedding. (Everett 2005: 628)

When he turns to another potential case of syntactic embedding (namely possession), he comments as follows:

> Neither the declarative nor the interrogative form of recursive possession is acceptable. No more than one possessor per noun phrase is ever allowed. [...] A cultural observation here is, I believe, important for understanding this restriction. Every Pirahã knows every other Pirahã, and they add the knowledge of newborns very quickly. Therefore one level of possessor is all that is ever needed. (Everett 2005: 630)

So, nested possessive structures (e.g., *Kó'oi's son's daughter*) are described as being 'recursive', thus implying (like Pinker and Jackendoff) that recursion = syntactic embedding (i.e., a constituent of one kind is embedded within a constituent of the same kind). [Note, though, Everett also implies that the level of embedding must be greater than 1 since he does not classify structures such as *Kó'oi's daughter* as being recursive despite the fact that they can be analysed as nested DPs].

6.3 Recursion in The Minimalist Program

'Recursion' may be undefined in HCF, but Chomsky did provide a definition (of some sort) in his 1995 monograph *The Minimalist Program*. The computational component of natural language (C_{HL}) is introduced, and 'the operations of C_{HL} recursively construct *syntactic objects*' (Chomsky 1995:226). Syntactic Objects (SOs) are defined as follows (see Chomsky 1995: 243):

> **Definition 6.1: Syntactic Objects**
>
> 1. lexical items
> 2. $K = \{\gamma\{\alpha, \beta\}\}$, where α, β are syntactic objects and γ is the label of K

So, all Lexical Items (LIs) are SOs, and further SOs can be created by combining existing SOs in a principled manner. Chomsky explicitly notes that it is clause 2 which provides the 'recursive step' (Chomsky 1995: 243).

An example: C_{HL} contains only one operation, (Merge), and $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ are all LIs in a given numeration, each having an index value of 1. One possible series of subsequent steps:

> **Example 1: Derivation**
>
> Given $\Sigma = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$:
>
> Step 1: $K_1 = \text{Merge}(\alpha_1, \alpha_2)$ and $\Sigma' = \{K_1, \alpha_3, \alpha_4\}$
> Step 2: $K_2 = \text{Merge}(K_1, \alpha_3)$ and $\Sigma'' = \{K_2, \alpha_4\}$
> Step 3: $K_3 = \text{Merge}(K_2, \alpha_4)$ and $\Sigma''' = \{K_3\}$

The derivation in Example 1 terminates when state Σ''' is reached because only one syntactic object, $\{K_3\}$, remains. Since 'no new objects are added [...] apart from rearrangements of lexical properties' (Chomsky 1995: 228) during the course of the derivation, the whole process is entirely determined by the operation Merge (which is applied 'recursively') and by the features associated with the LIs.

Presumably, Def.6.1 defines the 'recursive' procedure that HCF have in mind when they claim that recursion is central to FLN. If this is the case, then Pinker/Jackendoff/Everett and HCF are simply talking about different things since Def.6.1 does not *exclusively* create embedded structures (though it certainly creates these as well).

As a result, recent discussions concerning such matters are (understandably!) frequently confused and confusing. In a recent *Language Log* posting about Everett's work (for example), Mark Liberman has commented as follows:

> By "recursion", HCF mean "computational mechanisms ... providing the capacity to generate an infinite range of expressions from a finite set of elements". "Recursion" in this sense goes beyond the simple combinations of modifiers and heads ("red" + "cow" → "red cow"), or subjects and verbs ("Joan" + "disagree" → "Joan disagrees"), or any other construction that doesn't involve embedding a complex element repeatedly inside another element of the same type. Non-recursive constructions (like modifier+head) are very useful, and such embeddings multiply the set of messages that you can make out of a finite set of elements, but they don't "generate an infinite range of expressions" unless they operate recursively. (Liberman 2006)

The problem is that Def.6.1 does indeed generate 'the simple combinations of modifiers and heads'. In fact, given Def.6.1, modifier/head SOs (e.g., *red cow*) and SOs that contain embedded structures are generated in the same way: two SOs are Merged, and the features associated with the lexical items and functional categories determine the structure created.

7 Returning to Recursion (again)

SS is a frustrating text: it makes bold and intriguing claims, yet it does not substantiate them; important ideas and approaches are left vague and undefined – and the way in which 'recursion' is both discussed and implied in *SS* is typical:

- recursion is sometimes presented as being central, but it is never defined and, remains ambiguous
- various usages of recursive devices are not discussed (e.g., embedding conversions), though these are discussed (informally) in pre-*SS* texts
- it is claimed that recursive devices are sometimes helpful and sometimes unhelpful, but it does not indicate clearly when and why

Given the above, it is possible to claim the following:

- In 1957 'recursion' was a much used, much discussed, fundamentally ambiguous term that caused extensive misunderstandings
- In 2007 'recursion' is a much used, much discussed, fundamentally ambiguous term that causes extensive misunderstandings

During the last 50 years, the type of semi-formal linguistic theorising that *SS* popularised has certainly been influential!

There are several distinct, yet related, topics that cluster around the notion of 'recursion' in Generative Grammar (and other theories):

- Using finite components, a speaker-hearer of a natural language can produce a potentially infinite number of syntactic structures (e.g., *the old man*, *the old old man*, *the old old old man* etc) – this can be called **discrete infinity**
- natural languages combine smaller linguistic units in order to create larger linguistic units (e.g., [*old*] + [*man*] → [*old man*]) – this can be accomplished using a **constructional device**
- Certain linguistic structures of a particular type contain other linguistic structures of the same type (e.g., *I think that the old man laughs*, *Tom's sister's brother's wife*) – this can be called **syntactic embedding**

A few concluding thoughts:

- **discrete infinity** would not be possible if there were no **constructional device**
- **syntactic embedding** would not be possible if there were no **constructional device**
- **syntactic embedding** specifies a particular kind of syntactic structure (it makes no strong claims about the manner in which the structure was created)
- It is preferable to use terms such as **discrete infinity**, **constructional device**, and **syntactic embedding** rather than the term **recursion**

If the word 'recursion' can be avoided entirely, then perhaps it is time for linguists to stop talking about it altogether...

References

[1] Bar-Hillel, Y., (1953), 'On Recursive Definitions in Empirical Science', 11^{th} International Congress of Philosophy 5: 160-165.

[2] Bar-Hillel, Y., (1954), 'Logical Syntax and Semantics' Language 30: 230-237.

[3] Chomsky, N., (1951), *Morphophonemics of Modern Hebrew*, MA Thesis (published in 1979).

[4] Chomsky, N., (1953), 'Systems of Syntactic Analysis', The Journal of Symbolic Logic 18: 242-256.

[5] Chomsky, N., (1955), 'Logical Syntax and Semantics: their linguistic relevance', Language 31: 36-45.

[6] Chomsky, N., (1955b), 'Semantic Considerations in Grammar', Monograph Series in Language and Linguistics, 141-153.

[7] Chomsky, N., (1956), 'Three Models for the Description of Language', IRE Transactions of Information Theory, IT-2, 113-124.

[8] Chomsky, N., (1957), *Syntactic Structures*, Janua Linguarum, Mouton & Co.: 'S-Gravenhage.

[9] Chomsky, N., (1975[1955]), *The Logical Structure of Linguistic Theory*, The MIT Press: Cambridge, Massachusetts.

[10] Chomsky, N., (1995), *The Minimalist Program*, The MIT Press: Cambridge, Massachusetts.

[11] Church, A., (1936), 'An Unsolvable Problem of Elementary Number Theory', American Journalof Mathematics, 58: 345-363

[12] Dillinger, M. and Palácio, A., (1997), 'Generative Linguistics: Development and Perspectives. An interview with Noam Chomsky'. Chomsky no Brasil / Chomsky in Brazil: Revista de Documentacão de Estudos em Lingüística Teórica e Aplicada (D.E.L.T.A.) 13, N . Especial, 159-194.

[13] Everett, D.L., (2005), 'Cultural Constraints on Grammar and Cognition in Pirahã: Another Look at the Design Features of Human Language', Current Anthropology, Vol.46, Number 4, 621-646.

[14] Gödel, K., (1986a[1931]), 'Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme', reprinted in Gödel 1986c, Vol I, 144-194.

[15] Gödel, K., (1986b[1934]), 'On Undecidable Propositions of Formal Mathematical Systems', reprinted in Gödel 1986c, Vol I, 346-371.

[16] Gödel, K., (1986c), *Kurt Gödel: Collected Works*, Two Volumes, Oxford University Press: Oxford and New York.

[17] Harwood, F., (1955), 'Axiomatic Syntax: The Construction and Evaluation of a Syntactic Calculus', Language 31: 409-414.

[18] Hauser, M., Chomsky N., and Fitch, T., (2002), 'The Faculty of Language: What Is It, Who Has It, and How Did It Evolve', Science 298: 1569-1579.

[19] Kleene, S., (1952), *Introduction to Metamathematics*, North-Holland: Amsterdam.

[20] Lees, R.D., (1957), Review of Chomsky 1957, Language 33: 375-407.

[21] Liberman, M. 2006. 'Parataxis in Pirahã', May 19th, http://itre.cis.upenn.edu/m̃yl/ languagelog/archives/003162.html

[22] Matthews, P.H., (1993), *Grammatical Theory in the United States from Bloomfield to Chomsky*, Cambridge University Press: Cambridge.

[23] Newmeyer, F.J., (1980), *Linguistic Theory in America: the First Quarter-Century of Transformational Grammar*, Academic Press: New York, London, Toronto, Sydney, and San Fransisco.

[24] Peano, G., (1958[1889]), *Arithmetices Principia Nova Methodo Exposita*, reprinted in Unione Matematica Italiana (eds.), (1957-1959), *Opera scelte* 2: 20-55.

[25] Pike, K., (1952), 'More on Grammatical Prerequisites', Word 8: 106-21.

[26] Pinker, S. and Jackendoff, R., (2005), 'The Faculty of Language: What's Special about it?', Cognition 95: 201-236.

[27] Post, E.L., (1944), 'Recursively Enumerable Sets of Positive Integers and Their Decision Problems', Bulletin of the American Mathematical Society 50: 284-316.

[28] Shannon, C., and Weaver, W., (1949), *The Mathematical Theory of Communication*, University of Illinois Press: Urbana.

[29] Soare, R.I., (1996), 'Computability and Recursion', Bulletin of Symbolic Logic 2: 284-321.

[30] Tomalin, M., (2006), *Linguistics and the Formal Sciences: The Origins of Generative Grammar*, Cambridge University Press: Cambridge.

[31] Turing, A., (1936), 'On Computable Numbers, with an Application to the Entscheidungsproblem', Proceedings of the London Mathematical Society 42: 230-265.