

Discontinuous Structures in DBS and PSG

Roland Hausser
Universität Erlangen-Nürnberg (em.)
©Roland Hausser, January 3, 2021

Abstract

In linguistics, grammatical constructions with semantically connected word forms are called *discontinuous* if the word forms are not adjacent. For example, in (i) Peter looked the number up, the semantically related looked ... up are separated by the_number, and in (ii) Yesterday Mary danced the semantically related Yesterday ... danced are separated by Mary.

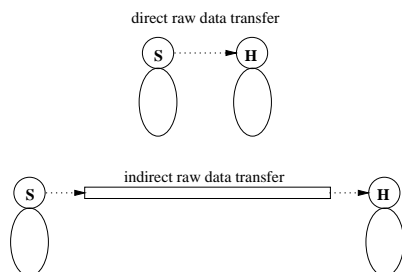
This paper explores why example (i) poses a descriptive problem for PSG (Phrase Structure Grammar), but not for DBS (Database Semantics), and why example (ii) poses a descriptive problem for DBS, but not for PSG. Then the PSG and DBS proposals for solving their respective problem are compared.

keywords: constituent structure, time-linear derivation, semantic relations

1 The Time-Linear Structure of Natural Language

Many syntactic-semantic content structures in human cognition are hierarchical, but the transfer of content in language communication is strictly linear. In the medium of speech, this holds for direct language communication, such as talking face to face, and indirect language communication, such as talking on the phone:

1.1 COMPARING FACE-TO-FACE WITH ON THE PHONE TALKING



The cognitive aspects are located inside the agents' heads, with **S** for speaker, **H** for hearer, and the arrow heads indicating direction. Transfer operates the same regardless of whether communication is face to face (direct raw data transfer) or talking on the phone (indirect raw data transfer). It is therefore not appropriate to build linguistic semiotics on information theory, as has been proposed by Eco¹ (1975).

All that is required of an artificial or natural transfer channel is the transmission of data without distortion (Shannon and Weaver 1948). However, though the transfer channel is not the place for reconstructing cognition, it poses a crucial structural requirement for language communication: the signs must be in a strictly linear order (canonized by de Saussure ([1916]1972) as his *second principe*). This is because humans can neither produce nor interpret several words, phrases, sentences,

¹Eco's most basic prototype of communication is a buoy "telling" an engineer the water level of a lake; the engineer's partner of discourse is mother nature. Grice's (1957) "bus bell model" and Dretsky's (1981) "doorbell model" have another human ringing the bell (CLaTR 2.2.4).

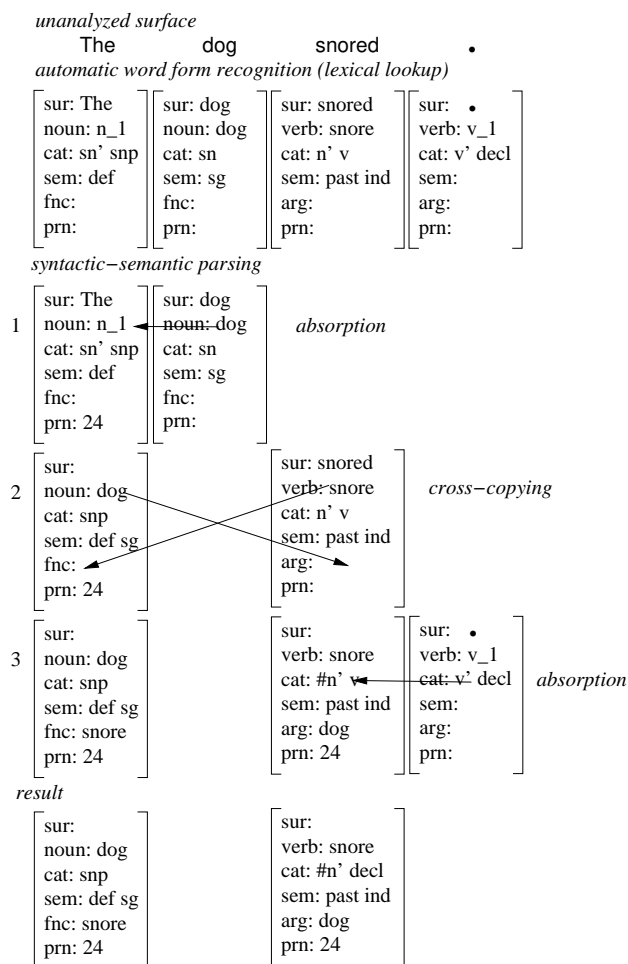
or texts simultaneously. Cognitive structures may be as hierarchical as necessary as long as language contents are coded in time-linear order fit for the transfer channel.²

In DBS, a content is a set (order-free) of proplets connected by address. Proplets are non-recursive feature structures with ordered attributes. As the basic units of propositions, proplets serve as the computational data structure. Language vs. nonlanguage contents differ in the presence vs. absence of language-dependent surfaces.

The speak mode takes a content as input and produces a surface sequence as output by traversing the input content along the address-coded semantic relations between proplets. The hear mode takes a surface sequence as input and produces an output content by concatenating proplets by address into a set. Because the speak and the hear mode take different input and output, they can not use the same algorithm.

Consider the hear mode taking *The dog snored.* as the input *surface* and the time-linear surface compositional derivation which produces the output *content*:

1.2 DERIVING CONTENT FROM SURFACE IN HEAR MODE



²Chomsky emphasizes that Generative Grammar (nativism) models the innate structure of natural language (structuralism), and is "not intended" for a transfer of content in language communication. It is unlikely, however, that the innate structure of natural language would do without a speak mode, a hear mode, and a transfer channel.

The input is four lexical proplets, i.e. unconnected and with nonempty sur slots. The output is a content of two nonlexical proplets, i.e. connected and with empty sur slots. The derivation order is (i) *time-linear* (left-associative³), shown by the stair-like addition of next word proplets, the analysis is (ii) *surface compositional* because each lexical item has a concrete sur value and there are no surfaces without a proplet analysis, and the activation and application of operations is (iii) *data-driven*.

Defining a content as a set of proplets connected by address is essential for storage in and retrieval from the agent's on-board memory (content-addressable database). The following example illustrates the order-free coding of the subject/predicate relation:

1.3 CONTENT OF The dog snored.

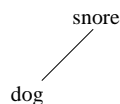
[sur: noun: dog cat: def sg sem: fnc: snore mdr: nc: pc: prn: 24]	[sur: verb: snore cat: #n' decl sem: past ind arg: dog mdr: nc: pc: prn:24]
--	--

Technically the subject/predicate relation is established by cross-copying.

Hierarchical semantic structure may be shown graphically as follows:

1.4 SEMANTIC RELATIONS UNDERLYING SPEAK MODE

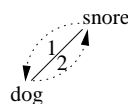
(i) SRG (semantic relations graph)



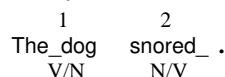
(ii) signature



(iii) NAG (numbered arcs graph)



(iv) surface realization



The (i) SRG and the (ii) signature on the left show the static aspect of the hierarchical content structure. The (iii) NAG (numbered arcs graph) and (iv) surface realization on the right show its dynamic traversal.

The arc numbers of the NAG are used for specifying (1) a think mode navigation and (2) a think-speak mode surface production as shown by the *surface realization*. The main intrapositional traversal operations in the think mode of DBS are (1) predicate/subject, (2) subject/ predicate, (3) predicate\object, (4) object\predicate, (5) noun\adnominal, (6) adnominal/noun, (7) verb\adverbial, (8) adverbial/verb, (9) noun→noun, (10) noun←noun, (11) verb→verb, (12) verb←verb, (13) adnominal→adnominal, and (14) adnominal←adnominal.

The (iv) *surface realization* shows language-dependent production. It is implemented as the speak mode riding piggy-back on the think mode navigation. The agent's memory provides the concepts of **dog** and **snore** as types (*declarative specification*). The agent's interface component provides raw data which are matched by types, resulting in tokens (*operational implementation*).

³Aho and Ullman (1977), p. 47; FoCL 10.1.1.

2 Constituent Structure Paradox of PSG

DBS and PSG differ ontologically in that DBS is agent-based data-driven, while PSG is sign-based substitution-driven. The derivations of PSG generate different language expressions from the same node, called **S** for sentence or start. The number of possible tree structures for a given surface grows exponentially with the length of the surface. Assuming binary branching, there are two PSG trees for a three-word sentence:

2.1 PSG TREES FOR THREE-WORD UNAMBIGUOUS SURFACE



From a formal point of view, both trees are equally well-formed.

However, because more than one tree for an unambiguous surface does not make sense linguistically, there must be an intuitive principle for choosing the “good” one. In chomskyan linguistics, this is the principle of Constituent Structure:⁴

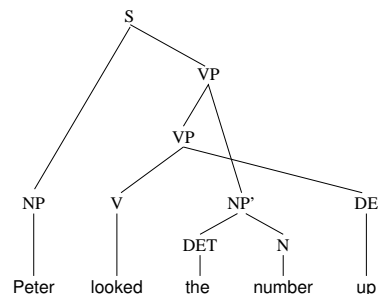
2.2 DEFINITION OF CONSTITUENT STRUCTURE

1. Words or constituents which belong together semantically must be dominated directly and exhaustively by a node.
2. The lines of a Constituent Structure may not cross (*nontangling condition*).

Assuming that **knows** and **John** belong closer together semantically than **Julia** and **knows**, the tree on the left satisfies 2.2 and is therefore considered linguistically correct, while the tree on the right is not. Yet it has been known at least since 1953 (Bar-Hillel 1964, p. 102) that there are certain natural language constructions with “discontinuous elements” which violate the definition of Constituent Structure.

Known as the Constituent Structure Paradox (FoCL Sect. 8.5), the problem may be illustrated with discontinuous **look__up** in the following attempts to analyze **Peter looked the number up** as a Constituent Structure:

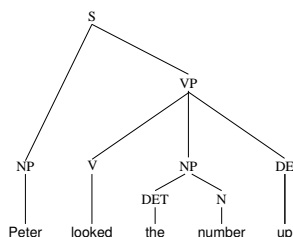
2.3 VIOLATING THE SECOND CONDITION OF 2.1



⁴Compared to the writings of Aristotle, Constituent Structure is quite recent. It evolved from the *immediate constituent analysis* of L. Bloomfield (1887–1949). Bloomfield’s student Z. Harris (1909–1992) turned the constituent analyses into substitution and movement tests. Harris’s student N. Chomsky turned the methodologically motivated tests into generative rules, called transformations and proffered as innate.

Here the semantically related expressions *looked* and *up* are dominated directly and exhaustively by a node, satisfying the first condition of 2.1. The analysis violates the second condition, however, because the lines cross.⁵

2.4 VIOLATING THE FIRST CONDITION OF 2.1



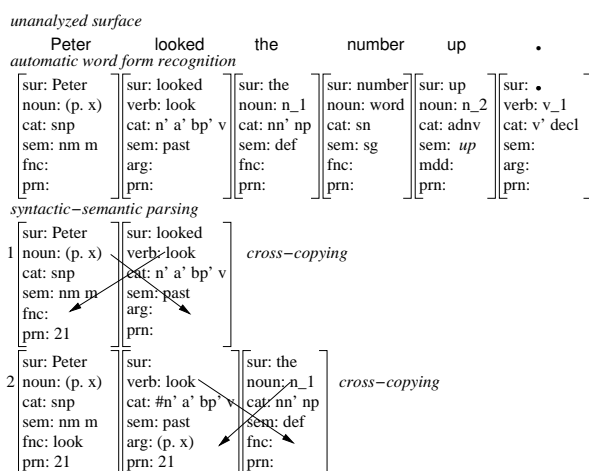
Here the lines of the tree do not cross, satisfying the second condition, but the semantically related expressions *looked* – *up*, or rather the nodes V and DE dominating them, are not *exhaustively* dominated by a node. Instead, the node directly dominating V and DE also dominates the NP *the_number*.

Undaunted by this empirical problem, Chomsky continued to use Constituent Structure for linguistically motivating the choice between possible phrase structure trees, the number of which grows exponentially with the length of unambiguous natural language surfaces (2.1). To also accommodate non-constituent structures, he added a transformation (movement) component on top of the context-free PSG base.

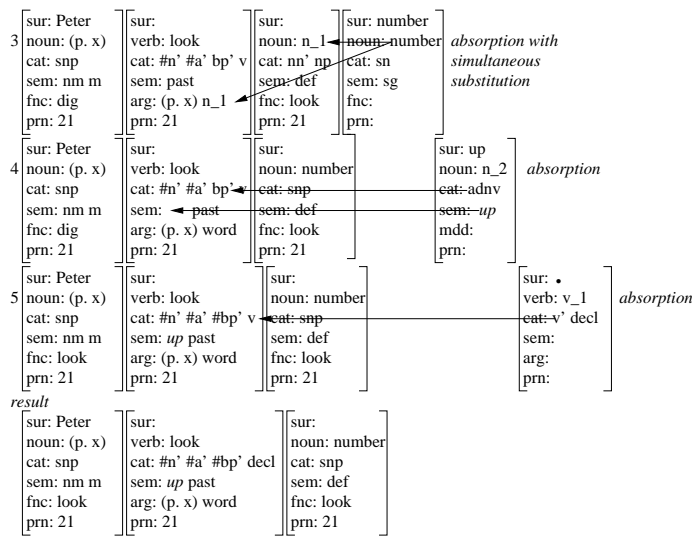
In summary, the output of the context-free base must satisfy Constituent Structure but can not accommodate some natural surfaces. The output of the transformations must satisfy the natural surfaces, but sometimes violate Constituent Structure. Worst of all, the introduction of transformations raises the computational complexity degree of context-free PSG polynomial (n^3) to undecidable (Peters&Ritchie 1973).

In DBS, the example shown in 2.3 and 2.4 for PSG derives without any problem in the strictly time-linear, surface-compositional, data-driven manner of the hear mode:

2.5 GRAPHICAL HEAR MODE DERIVATION IN DBS



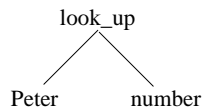
⁵DPSG (Discontinuous Phrase Structure Grammar, Bunt et al. 1987) argues for accepting crossing lines in Phrase Structure Trees (quasi three-dimensional). It was preceded by pleas for using only context-free phrase structure by Harman (1963), McCawley (1982), Gazdar et al. (1985), and others.



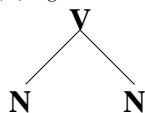
The semantic relations in the resulting content are shown by the following graph analysis. The surface realization of the speak mode is English:

2.6 GRAPH ANALYSIS UNDERLYING PRODUCTION IN DBS

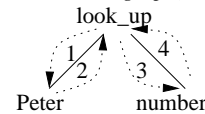
(i) SRG (semantic relations graph)



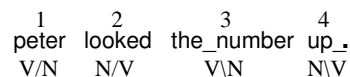
(ii) signature



(iii) NAG (numbered arcs graph)



(iv) surface realization



In the content derived in 2.5, the discontinuous element **up** is coded as the initial *sem* value of the *look* proplet in line 4. In the (iv) *surface realization*, **up_.** is realized from the finite verb (goal proplet of arc 4 in the (iii) *NAG*).

3 Suspension in DBS

The reason why **Peter looked the number up** violates Constituent Structure but **Yesterday Mary danced** does not may be shown by comparing their respective context-free PSG derivations, based on the possible substitutions of rewrite rules:

3.1 CONTEXT-FREE PSGS FOR DISCONTINUOUS STRUCTURES

Peter looked the number up

- S → NP VP
- VP → VP NP
- VP → V NP DE
- NP → N
- NP → DET N
- N → Peter
- V → looked

Yesterday Mary danced

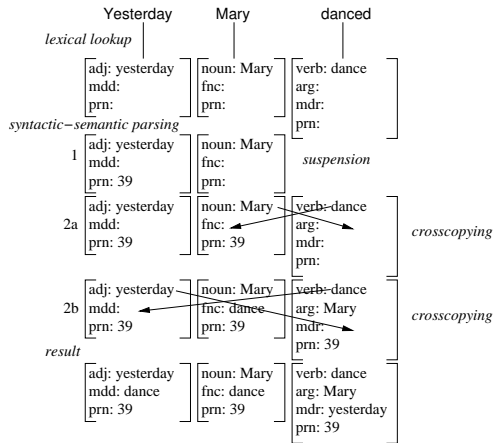
- S → ADV VP
- VP → NP V
- NP → N
- ADV → yesterday
- N → Mary
- V → danced

DET → the
 N → number
 DE → up

The left PSG for 2.4 violates the requirement of *exhaustive* dominance with the rule $VP \rightarrow V NP DE$.⁶ The PSG on the right, in contrast, generates Yesterday Mary danced without any problem: first the ADV is placed in initial position by $S \rightarrow ADV VP$; then $VP \rightarrow NP V$ places the subject noun after the ADV and before the V.

The apparent problem of DBS with Yesterday Mary danced, in contrast, results from the time-linear derivation order. It creates a temporary situation in which the elementary adverb yesterday can not be connected because the semantically related verb has not yet arrived. The solution is a *suspension* until danced becomes available:

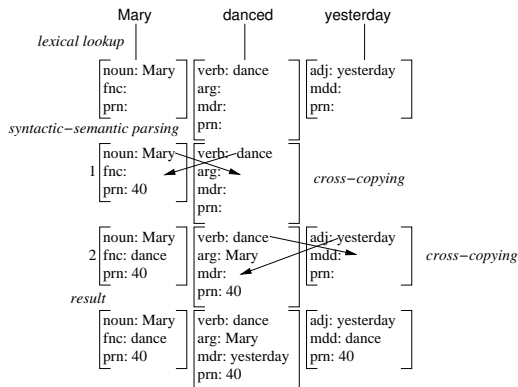
3.2 HEAR MODE DERIVATION OF Yesterday Mary danced.



Because operations are triggered by the input data (data-driven), instances of suspension are handled automatically, without any need for extra software.

The phenomenon is asymmetric in that neither suspension nor absorption is required by the semantically equivalent word order Mary danced+yesterday.

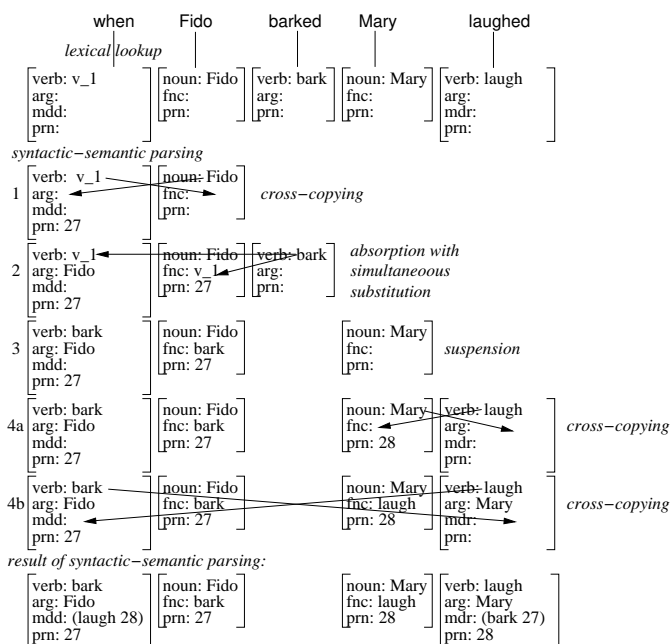
3.3 HEAR MODE DERIVATION OF Mary danced yesterday.



⁶Perhaps surprisingly, there exists a context-free PSG analysis for Peter looked the number up which satisfies 2.2. It is shown in 5.1 on the left, opposite the tree for Yesterday Mary danced.

Suspension at the elementary level scales up directly to the phrasal and clausal levels. At the clausal level, the relation is extrapositional:

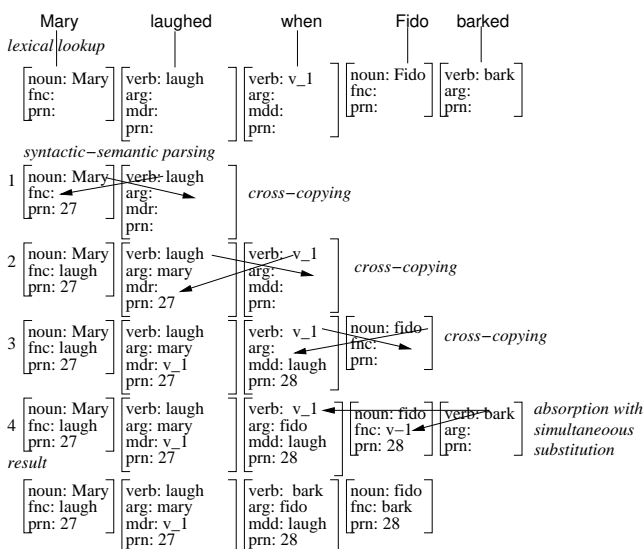
3.4 INTERPRETATION OF When Fido barked Mary laughed



The suspension occurs in line 3 and is compensated in lines 4a and 4b.

If the optional modifier clause follows the main clause, there is no suspension, just as in 3.3:

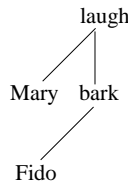
3.5 INTERPRETATION OF Mary laughed when Fido barked



Consider the semantic relations graph which underlies both surfaces:

3.6 SEMANTIC RELATIONS UNDERLYING SPEAK MODE PRODUCTION

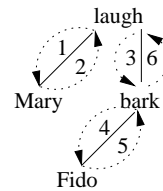
(i) SRG (semantic relations graph)



(ii) signature



(iii) NAG (numbered arcs graph)



(iv) surface realization

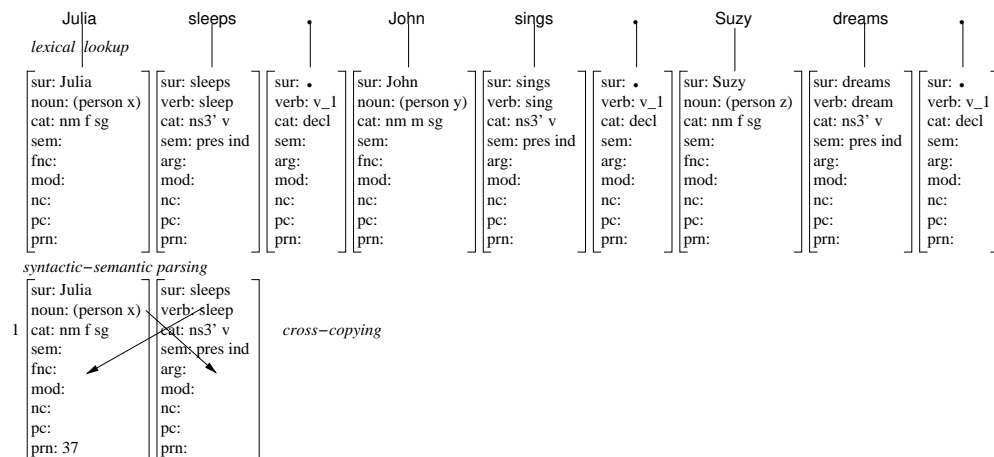
	3	4	5	6	1	2
(a)	When	Fido	barked	Mary	laughed	•
	V V	V/N	N/V	V V	V/N	N/V
	1	2	3	4	5	6
(b)	Mary	laughed	when	Fido	barked	•
	V/N	N/V	V V	V/N	N/V	V V

The suspension at the elementary level 3.2 and its variant 3.4 at the clausal level of grammatical complexity show that there are natural surface orders in which suspension cannot be avoided. Regarding the linear complexity of DBS, the ambiguities induced suspensions are benign because they are not recursive (TCS).

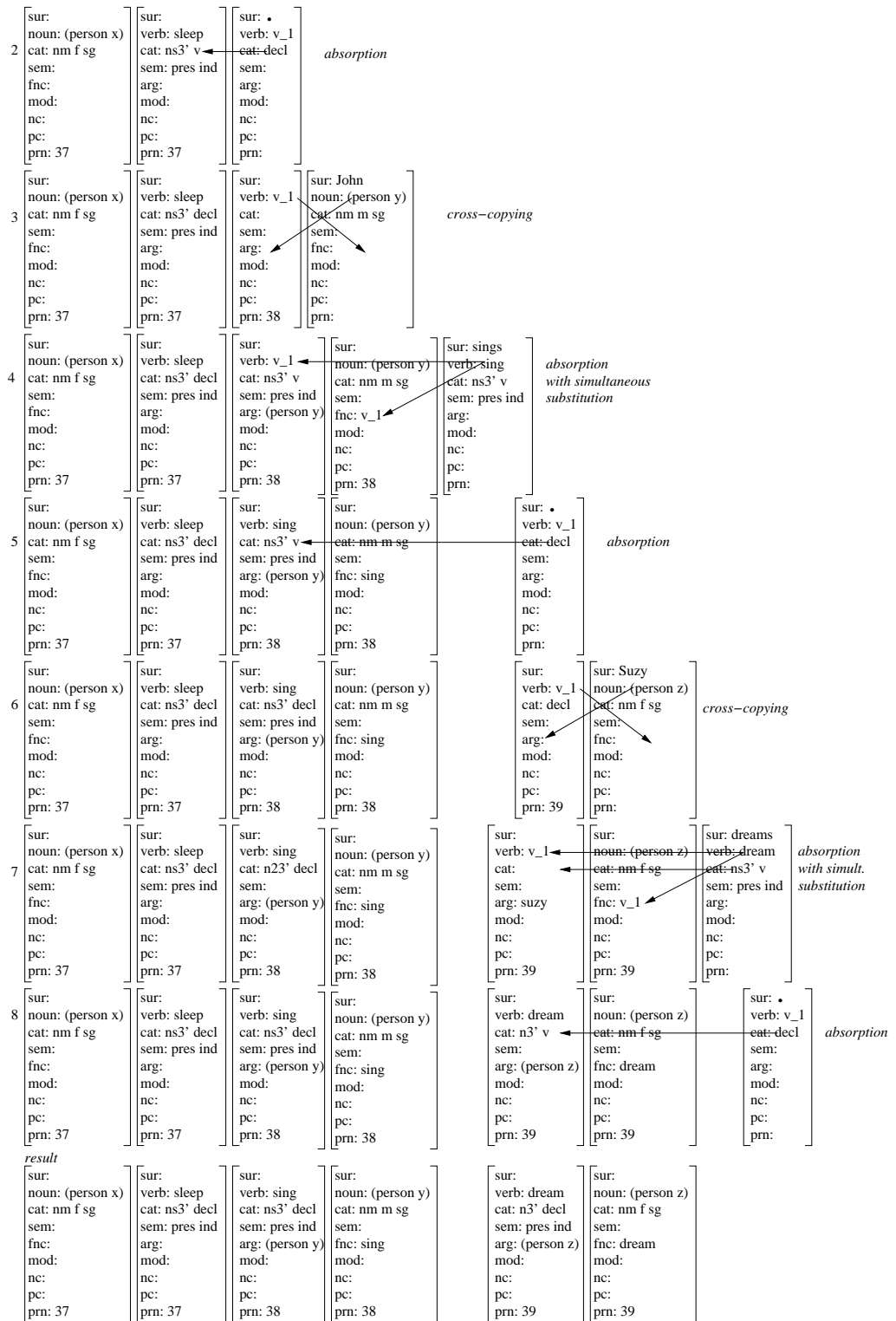
It seems that discontinuous filler-slot constructions require a suspension (4.1), whereas discontinuous slot-filler constructions make do with an absorption (i.e. without ambiguity, 4.2). An example is extrapositional coordination, as in a text. The relation between two adjacent sentences holds between the top verb of sentence n (into which the interpunctation has been absorbed) and the top verb of sentence $n+1$. In a sequence of English declaratives, for example, the verb of sentence $n+1$ is usually preceded by the subject or an adverbial, creating the discontinuity.

The pivot of the absorption transition from the verb of sentence n to the verb of sentence $n+1$ is the interpunctation (function word). This is shown by the following the hear mode derivation⁷ of Julia sleeps. John sings. Suzy dreams.:

3.7 ABSORPTIONS IN EXTRAPROPOSITIONAL COORDINATION



⁷For the earlier suspension analyses see NLC Sects. 11.5 (hear mode) and 11.6 (speak mode).



Despite two extrapositional transitions with intervening subjects (lines 3 and 6),

there are no suspensions and no additional operation applications. Instead, the full stop proplets in lines 2 and 5 simultaneously (i) supply the verbal mood **decl** to the verb of proposition *n*, i.e. the one the full-stop belongs to, and (ii) cross-copy with the subject of proposition *n+1* if there is one. If there is no subsequent proposition, in contrast, the proplet of the interpunctation is discarded once its verbal mood value has been utilized (line 8).

4 Discontinuity with and without Suspension in DBS

The domain-range structure of a semantic relation may be viewed as a filler-slot constellation, with the domain providing the filler (also called the argument, actant, or complement) and the range (also called functor, codomain, or valency⁸ carrier) providing the slot. In (i) subject/predicate, the subject is the filler and the predicate provides the slot, in (ii) object/predicate, the predicate provides the slot and the object is the filler, in (iii) modifier/modified, the modifier is the filler and the modified provides the slot, and in (iv) coordination, the slot-filler relation between conjuncts is bidirectional.⁹

If filler and slot provider are adjacent in the hear mode, i.e. if there are no intervening items, order does not make a difference. However, if there are intervening items and the slot provider precedes, then (a) the slot defines the kind of compatible filler, (b) the relation is initiated, and (c) the hearer can simply wait until the filler arrives and plops into place. If the filler precedes, in contrast, there may be many kinds of slot providers which is why no relation can be initiated; instead the hearer must wait until automatic word form recognition provides the slot, finally enabling an on the spot filler-slot combination.

In English filler-slot constellations, intervening items must be bridged by a suspension. Consider the following examples:

4.1 SUSPENSION IN ‘FILLER PRECEDES SLOT’ DISCONTINUITIES

1. Clausal subject precedes main clause.
That Fido found a bone surprised Mary.¹⁰
2. Adverbial precedes predicate (3.2)
Yesterday Mary danced.
3. Clausal modifier precedes main clause (3.4)
When Fido barked Mary laughed.
4. Subject gapping (TExer, Sect. 5.2)
Bob bought an apple, peeled a pear, and ate a peach.
5. Predicate gapping (TExer, Sect. 5.3)
Bob bought an apple, Jim a pair, and Bill a peach.
6. Long distance dependency (TExer, Sect. 5.5)
Who(m) did John say that Bill believes that Mary loves?

The grammatical role of the function words *that*, *when*, and *and* is shown in detail in the explicit hear mode derivations referred to.

⁸L. Tesnière (1959).

⁹This explicit specification of the semantic relations of structure is in contrast to the “belong semantically together” notion of the Constituent Structure definition 2.2.

¹⁰In *That Fido barked amused Mary* (TExer, Sect. 2.5) the V/V relation between *bark* and *amuse* is not discontinuous and no suspension is needed. It is similar in the V/A relation of *Mary danced yesterday* (3.3).

Discontinuous ‘slot provider precedes filler’ constructions, in contrast, do not require a suspension. Instead, an *absorption* suffices:

4.2 NO SUSPENSION IN ‘SLOT PRECEDES FILLER’ DISCONTINUITIES

1. Verb precedes bare preposition (2.5)
Peter looked the number up.
2. Clausal object (TExer, Sect. 2.6)
Mary heard that Fido barked.
3. Main clause precedes clausal modifier (3.5)
Mary laughed when Fido barked.
4. Object gapping (TExer, Sect. 5.4)
Bill bought, Jim peeled, and Bill ate a peach.
5. Repeating object clauses (TExer, Sect. 5.6)
Mary saw the man who loves the woman who fed Fido.
6. Period precedes subject in extrapositional coordination (3.7):
Julia sleeps. John sings. Suzy dreams.

In English, ‘slot provider precedes filler’ discontinuities use the additional function words *who*, *whom*, and *.* to (i) cross-copy with the intervening item and then to (ii) absorb and simultaneously substitute the filler.

In gapping constructions, the relation between the order of filler and slot provider, and the use of suspension vs. absorption is the reverse as compared to 4.1 and 4.2:

4.3 ABSORPTION VS. SUSPENSION IN GAPPING

1. Subject gapping (TExer, Sect. 5.2)
Bob bought an apple, peeled a pear, and ate a peach.
Exception: filler-slot without suspension
2. Predicate gapping (TExer, Sect. 5.3)
Bob bought an apple, Jim a pair, and Bill a peach.
Exception: filler-slot without suspension
3. Object gapping (TExer, Sect. 5.4)
Bill bought, Jim peeled, and Bill ate a peach.
Exception: slot-filler with suspension

Object gapping is special in that it requires not only suspension, but also a derivation-external cache for storing the gap providers until the filler (as the standard location of the gap list) arrives.

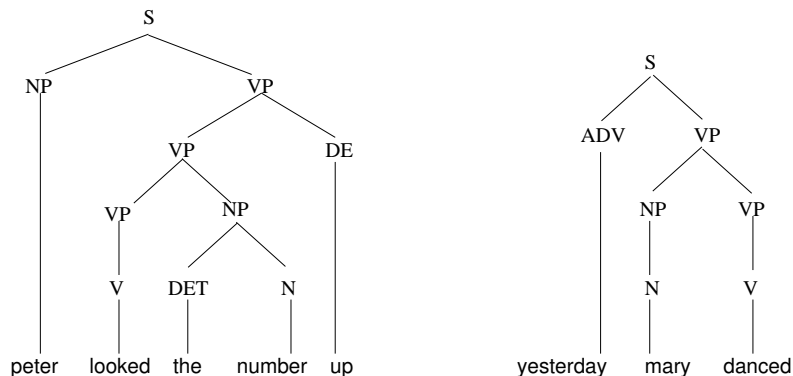
5 Conclusion

This paper started with an enigma. The examples

- (i) *Peter looked the number up.* and
- (ii) *Yesterday Maria danced.*

are both discontinuous. For context-free PSG, (i) poses a problem, solved by adding a movement component, while (ii) runs straight through. For DBS, (i) runs straight through, while (ii) poses a problem, solved by suspension and data-driven cross-copying (3.2). Let us conclude by showing what applying the PSG solution for example (ii) to example (i) in 3.1 would look like:

5.1 EXAMPLE 2.3 AS A PHRASE STRUCTURE SATISFYING 2.2



If the phrase structure on the right satisfies the Constituent Structure definition 2.2 then so does the one on the left: in both trees the lines do not cross; also the rules (a) $VP \rightarrow VP\ DE$ used on the left and (b) $S \rightarrow ADV\ VP$ used on the right are alike in that two nodes which “belong together semantically” are dominated directly and exhaustively by a single node.¹¹

Today’s PSG interprets “belonging together semantically” as the functor-argument relation of symbolic logic (first order predicate calculus). Thereby the question of whether the functor precedes or follows in the surface is treated as the language-dependent “problem of linearization.” In data-driven DBS, in contrast, the functor-argument relation introduces a surface-compositional asymmetry, namely between a *slot-filler* and a *filler-slot* relation.

A slot provider looking for a filler is like trying a bag of old keys on a given piece of furniture, while a filler looking for a slot provider is like trying a key on an open number of furniture pieces. In English declaratives, the potential inefficiency of the filler-slot order in the subject/predicate constellation is effectively compensated by the post-nominative position (FoCL Sect. 18.3) of the finite verb, which provides adjacency.

Acknowledgement: Thanks to Professor Kiyong Lee for helpful comments.

Bibliography

- Aho, A.V. and J.D. Ullman (1977) *Principles of Compiler Design*, Reading, Mass.: Addison-Wesley
- Bar-Hillel, Y. (1964) *Language and Information. Selected Essays on Their Theory and Application*, Reading, Mass.: Addison-Wesley

¹¹What is so wrong with the phrase structure on the left that adding a movement component was preferred even at the price of making the grammar algorithm undecidable? First, being undecidable was fashionable in substitution-driven complexity analysis of the time, e.g. Post’s (1946) Correspondence Problem. Second, discontinuities were not the main motivation for introducing a movement component. Instead it was the assumption of a universal, innate context-free base in combination with a transformation component (nativism). Thus a discontinuity analysis exceeding the generative power of context-free PSGs came in handy as “empirical” support for the Standard Theory (ST, Chomsky 1965). The undecidability of Transformational Grammar was formally proven in active consultation with Professor Chomsky.

- Bloomfield, L. (1933) *Language*, New York: Holt, Rinehart, and Winston
- Bunt, H., J. Thesingh, and K. van der Sloot (1987) “Discontinuous Constituents in Trees, Rules, and Parsing,” in *Proc. 3rd Conference of the European chapter of the ACL*, pp. 203-210. Copenhagen, DK, Univ. of Copenhagen
- CC = Hausser, R. (2019) *Computational Cognition; Integrated DBS Software Design for Data-Driven Cognitive Processing*, DOI: 10.13140/RG.2.2.33658.85440, lagrammar.net
- Chomsky, N. (1965) *Aspects of the Theory of Syntax*, Cambridge, Mass.: MIT Press
- CLaTR = Hausser, R. (2011) *Computational Linguistics and Talking Robots; Processing Content in DBS*, pp. 286. Springer (preprint 2nd ed. lagrammar.net)
- Dretske, F. (1981) *Knowledge and the Flow of Information*, Cambridge, Mass.: Bradford Books/MIT Press
- Eco, U. (1975) *A Theory of Semiotics*, Bloomington, Indiana: Indiana Univ. Press
- FoCL = Hausser, R. (1999) *Foundations of Computational Linguistics*, 3rd ed. 2014, Springer
- Gazdar, G., E. Klein, G. Pullum, and I. Sag (1985) *Generalized Phrase Structure Grammar*. Cambridge, MA: Harvard Univ. Press
- Grice, P. (1957) “Meaning,” *Philosophical Review*, Vol. 66:377–388
- Harman, G. (1963) “Generative Grammar without Transformational Rules: a Defense of Phrase Structure,” *Language*, Vol. 39:597–616
- Harris, Z. ([1941]1951) *Methods in Structural Linguistics*, Chicago: Univ. of Chicago
- McCawley, J.D. (1982) “Parentheticals and Discontinuous Constituent Structure,” *Linguistic Inquiry*, Vol. 13.1:91-106
- NLC = Hausser, R. (2006) *A Computational Model of Natural Language Communication: Interpretation, Inference, and Production in DBS*, pp. 360. Springer (preprint 2nd ed. lagrammar.net)
- Peters, S., and R. Ritchie (1973) “On the Generative Power of Transformational Grammar,” *Information and Control*, Vol. 18:483–501
- Post, E. L. (1946) “A variant of a recursively unsolvable problem,” *Bull. Amer. Math. Soc.* Vol. 52:264–269
- Saussure, F. de [1916](1972) *Cours de linguistique générale*, Édition critique préparée par Tullio de Mauro, Paris: Éditions Payot
- Shannon, C.E., and W. Weaver (1949) *The Mathematical Theory of Communication*, Urbana, Illinois: University of Illinois Press
- TCS = Hausser, R. (1992) “Complexity in Left-Associative Grammar,” *Theoretical Computer Science*, Vol. 106.2:283-308
- Tesnière, L. (1959) *Éléments de syntaxe structurale*, Paris: Editions Klincksieck
- TExer3 = Hausser, R. (2020) *Twentyfour Exercises in Linguistic Analysis, DBS software design for the Hear and the Speak mode of a Talking Robot* (2nd ed. lagrammar.net)