Comments on Pullum

AD NEELEMAN

1. Marr's Three Levels of Description

My comments on Pullum's article will follow after a considerable detour. But for those who want to know where I am headed, let me summarize my criticism in a sentence. In my view, several of the arguments that Pullum presents in favour of a constraint-based model of grammar are weakened considerably by the fact that not enough consideration is given to the distinction between competence and performance.

I begin with the basics. Marr (1982) argues that any machine carrying out an information-processing task must be understood at three levels of description. Marr calls these levels *computational*, *algorithmic*, and *implementational*. The computational level of description is the most abstract. It gives an account of the device in terms of the logical structure of the mapping that it carries out from one type of information to another. If we are talking about a cash register (which is Marr's example), one of the things it does is addition:

(1) $+$

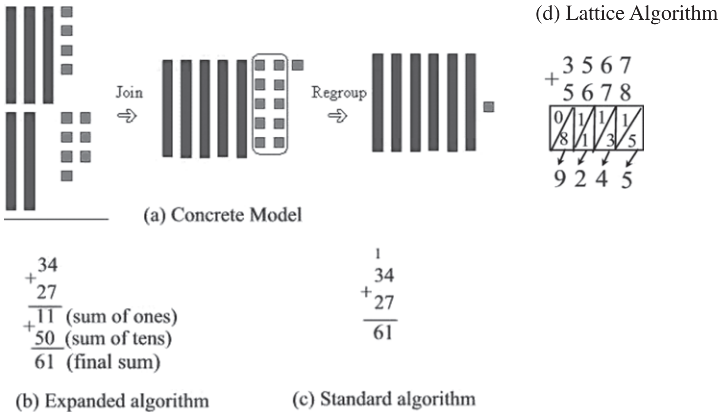
The next level down is a description of the algorithm that yields the desired input–output mapping. I do not know what algorithms a cash register uses, but in (2) I give four examples of algorithms that people use for addition. I will not discuss details, as the pictures are largely self-explanatory. The point is simply that there can be numerous algorithmic instantiations of a function like (1).[1]

Address for correspondence: Department of Linguistics, Room 110, Chandler House, 2 Wakefield Street, London WC1N 1PF, UK.
Email: a.neeleman@ucl.ac.uk

[1] The importance of a workable algorithm becomes apparent when one uses roman numerals to add 431 (CDXXXI) to 1723 (MDCCXXIII).

Mind & Language, Vol. 28, No. 4 September 2013, pp. 522–531.

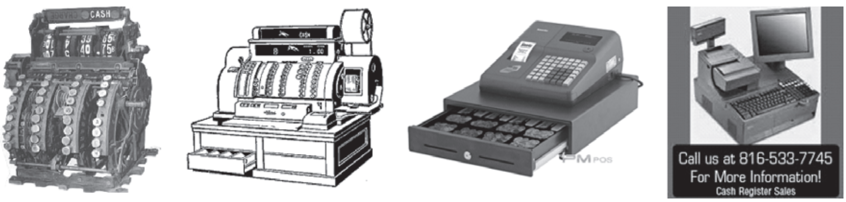
(2)



(Found at http://faculty.atu.edu/mfinan/2033/section12.pdf)

At the most basic level, we must offer a description of how the algorithm and its input and output are realized physically. It will be clear that there are potentially many physical implementations of any given algorithm. Below are a few pictures of cash machines to illustrate this; it is conceivable that the machines depicted all carry out a very similar algorithm using very different hardware.

(3)

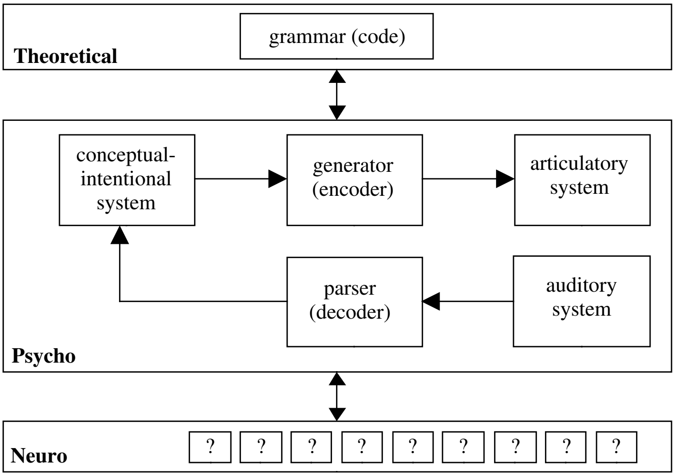


The crucial fact to be recognized is that the relation between Marr's three levels of description need not be transparent (the same function can map to many algorithms, which in turn can map to many physical implementations). This means that a full understanding of a computational device requires a description at all three levels.

2. How Does Linguistics Fit In?

The human language faculty is of course a computational device. The study of language should therefore contribute descriptions at the computational, algorithmic and implementational level. These map naturally onto three subdisciplines: theoretical linguistics, psycholinguistics and neurolinguistics. Theoretical linguistics deals with knowledge of language; that is, it gives a description of the formal properties of the way in which language associates sound and meaning. Psycholinguistics deals with the algorithms that implement knowledge of language in speech production and speech perception (among many other things). Neurolinguistics deals with the hardware implementation of these algorithms in the brain. The following diagram from Neeleman and Van de Koot, 2010 might be helpful in clarifying the overall picture (see that paper for details):

(4)



This characterization of the task that linguistics faces is of course not new. As the quote in (5) demonstrates, it is how Marr interprets the enterprise. In this quote Marr presupposes some form of generative grammar (referred to as 'Chomsky's theory'), but of course the conclusion reached is independent of the exact computational description of the language faculty one adopts (that is, it is true, whether the syntax is constraint-based or modelled as a generative procedure).

(5) [F]inding algorithms by which Chomsky's theory may be implemented is a completely different endeavor from formulating the theory itself. In our terms, it is a study at a different level, and both tasks have to be done (Marr, 1982, p. 28).

In fact, Chomsky argued already in 1965 that theoretical linguistics is distinct from the study of linguistic algorithms and their neural implementation. The terms he used to characterize the distinction are *competence* (as described by linguistic theory) and *performance* (as described, much later, by psycho- and neuro-linguistics). This point is hammered home very early on in *Aspects of the Theory of Syntax*:

(6) Linguistic theory is concerned primarily with an ideal speaker-listener, in a completely homogeneous speech-community, who knows its language perfectly and is unaffected by such grammatically irrelevant conditions as memory limitations, distractions, shifts of attention and interest, and errors (random or characteristic) in applying his knowledge of the language in actual performance (Chomsky, 1965, p. 3).

(7) When we say that a sentence has a certain derivation with respect to a particular generative grammar, we say nothing about how the speaker or hearer might proceed, in some practical or efficient way, to construct such a derivation. These questions belong to the theory of language use—the theory of performance (Chomsky, 1965, p. 9).

Although some of the clarity of *Aspects* has been lost along the way, I still believe that the only coherent interpretation of recent syntactic theorizing (including theorizing in the *Minimalist Program*) crucially relies on the competence-performance distinction (and hence on Marr's view of what it means to study a computational device).

3. Constraints and Procedures

The central conclusion of Pullum's article is that it is better to conceive of syntactic theory as a set of constraints (MTS) than as a generative procedure (GES). At first sight, this seems to neatly match the model in (4). After all, constraints are static and are therefore most naturally conceived of as belonging to the most abstract level of description. Given that parsing and generation of linguistic utterances must be procedural, one might be inclined to think of procedures as belonging to the intermediate level of description.

Indeed, a parser that makes use of constraints to filter out ill-formed structures generated by a minimally specified structure builder is unlikely to be successful. The structure builder will be able to generate a very large number of structures, the vast majority of which will have to be rejected. The effort required to deal with these ill-formed structures will quickly overwhelm the parsing process (for relevant discussion, see Marcus, 1980; Berwick and Weinberg, 1984; Abney, 1991; and Kolb and Thiersch, 1991). Here's a toy example from Neeleman and Van de Koot, 2010 which illustrates the growth of candidate structures for a given input depending on how many grammatical constraints are incorporated in the structure builder:

(8)

	A	***B***	***C***	***D***	***E***
Men (1 word)	1	1	1	2	1
Men slept (2 words)	2	2	1	4	1
The man slept (3 words)	12	8	2	8	1
Men bought a book (4 words)	112	40	5	16	1
A man bought a book (5 words)	1360	224	14	32	1
Men said men bought a book (6 words)	19872	1344	42	64	1
Men said a man bought a book (7 words)	?	8448	132	128	1

A: merge with weak inclusiveness and binary branching built in
B: merge with strong inclusiveness and binary branching built in
C: merge with strong inclusiveness, binary branching and label assignment under directionality built in
D: 2^n
E: merge with strong inclusiveness, binary branching, label assignment under directionality and selection built in

It is not important to know the nature of the grammatical constraints referred to ('weak inclusiveness', 'binary branching', etc.). These are discussed in some detail in Neeleman and Van de Koot, 2010, for those who have an interest in such matters. What is important is that the parsing process looks likely to succeed only if almost all relevant grammatical constraints are incorporated in the structure builder (as in column E). Therefore, if constraints have a place anywhere in linguistics, they must be part of linguistic theory / the theory of competence / the computational level of description.

However, neither the fact that a constraint-based account of the language faculty must belong to the computational level of description, nor the fact that an account at the algorithmic level of description must be procedural, implies that an account at the computational level must be non-procedural. It is possible that the most insightful description of the formal properties of the way in which language associates sound and meaning is in terms of a generative procedure. All that the model in (4) tells us is that such a procedure would not have to map transparently onto the procedures that are relevant to parsing and generation. We should let the facts decide between these different views. But what facts?

4. Native Speaker Judgments

The main data used to test linguistic theories consist of native speaker intuitions concerning the grammaticality of test sentences. For example, if I want to find out whether English is an SOV or SVO language, I may ask a native speaker to judge the following:

(9) a. John the book bought.
  b. John bought the book.

It is tempting to treat grammaticality judgments as directly providing evidence about the grammar of a language. However, this would be a fallacy. Grammaticality judgments do not directly probe knowledge of language. A speaker judging the acceptability of a sentence must use the hardware at their disposal to parse incoming speech sounds. As the parsing algorithms implement the grammar, the experience of ungrammaticality could be seen as one in which the parser is unable to assign a single connected structure to a given string. In other words, grammaticality judgments are a performance phenomenon used by theoretical linguists to make decisions about matters of competence. We must therefore be careful in our interpretation of the 'data'. What we think of as data are not simply facts about grammatical competence.

It stands to reason that the parser will have evolved to be as fast and as robust as possible. Both properties can give rise to misleading results, in that they may lead to twists in the relationship between grammaticality at the computational level and acceptability as experienced by subjects asked for their intuitions.

For example, it is well known that certain grammatical errors in subject-verb agreement tend to escape the attention of subjects when a different NP with appropriate features appears in the vicinity of the verb (as in the ungrammatical example in (10); see Wagers, Lau and Phillips, 2009). This is presumably the result of a performance system that favours speed over accuracy.

(10) *The key to the cabinets unsurprisingly were rusty from years of disuse.

Robustness in parsing means that a hearer may be able to assign an interpretation to an input for which no complete structure exists according to the rules of grammar. This could be the result of discourse-level processes, but for certain ill-formed sentences it is feasible that the performance mechanisms carry out repairs. A low-level example of such a repair is phoneme restoration[2] (Warren, 1970). It seems likely to me that there are higher-level repairs that locally affect phoneme, morpheme or constituent order, among other things. In other words, while some strings that are ungrammatical (in that they cannot be covered by a single connected structure) will be beyond salvation, other ungrammatical strings can probably be assigned a single connected structure, albeit at the cost of extra parsing efforts required for repairs. This is true, whether knowledge of language is described in terms of constraints or procedures.

In all likelihood, the reverse situation also exists. Some perfectly grammatical structures are hard to parse (for instance because they give rise to garden path effects), or in fact impossible to parse. It has been argued that certain cases of multiple centre embedding may fall in the latter category (that is, the parser is held responsible for the unacceptability of examples like *the cheese that the mouse that the cat caught ate was imported from France*).

Thus, hypotheses about the grammar do not directly predict grammaticality judgments. They predict grammaticality judgments in conjunction with a theory of performance about which relatively little is known. For reasons of practicality, linguists must therefore work under some idealization of the influence of the performance systems on grammaticality judgments.

The standard assumption is that this influence is negligible (compare the quote in (6)). However, some linguistics have made other assumptions that are usually not explicitly presented as such and that assume some degree of transparency between the descriptions of the language faculty at the computational and descriptive levels. The idea is that, at least in some cases, there is systematic association between the level of experienced unacceptability and the grammatical principle that is violated.

---

[2] In Warren's own words:

> In 1970, I reported that when a speech sound or an entire syllable in a sentence was deleted and replaced by a louder noise such as a cough, the sentence seemed intact, and even when listeners were informed that a portion of the sentence was deleted and replaced by a cough, they could not identify the missing speech sounds nor could they locate the extraneous sound's position in the sentence (http://recherche.ircam.fr/equipes/pcm/cheveign/sh/keele/warren/warren.html).

For example, in movement theory, violations of Subjacency are taken to give rise to weak unacceptability, as compared to violations of the Empty Category Principle (see Chomsky, 1986, among others). If correct, this could be explained in a model like (4) by saying that there is a repair strategy for (certain) Subjacency violations, but not for ECP violations, leading to subjects having different experiences of two types of examples that are both ungrammatical at the computational level of description.

While this is a perfectly reasonable way to proceed, the reality is that we cannot know before hand where the data reflect properties of the grammar (as instantiated in the performance systems) and where they reflect aspects of the performance systems that have to do with efficiency and robustness. In other words, in testing hypotheses about the grammar additional hypotheses about the implementation of the grammar in the performance systems must be made, and what is tested is this constellation of hypotheses, rather than grammatical theory on its own. There is nothing particularly remarkable about this, except perhaps that it is rarely commented on in the literature.

5. Pullum's Arguments Against GES and in Favour of MTS

It seems to me that the issue explored in the previous sections is particularly relevant to the arguments that Pullum presents in favour of a constraint-based grammar (or a model-theoretic syntax, to be more precise). These arguments are built on general properties of linguistic data that—in the absence of a thorough analysis of individual examples—may not reflect the grammar itself, but could just as well be contributed to limitations of the performance systems as a tool for generating grammaticality judgments.

I first consider the argument from *gradience of ungrammaticality*. The idea is that the standard derivational model of grammar cannot deal with the fact that some sentences are neither fully acceptable, nor fully unacceptable. This is because a generative procedure, by its very nature, can either produce a string or not produce a string. There is no middle way. A constraint-based grammar, however, allows strings that violate certain constrains, but not others.

If the observable data were direct expressions of competence grammar, this would be a very strong argument. But as argued above, the data are not. A subject's experience of a test sentence does of course depend on whether that sentence is grammatical (that is, whether in parsing it can be covered by a single connected structure). However, to repeat, it also depends on how easy it is to find a covering structure (if one exists) and how easy errors are detected (if the sentence is ungrammatical). If no covering structure is found, the subject's experience will further be affected by whether or not the substructures the parser builds up can be integrated though discourse mechanisms and whether or not there are repair mechanisms that can fix the problem. Given this range of factors, we expect, irrespective of the nature of the competence grammar, to find variation in grammaticality judgments. In particular, gradience of judgments is perfectly compatible with a description of knowledge of language in terms of a generative procedure. As the example of Subjacency violations versus ECP violations in the previous section demonstrated, this is true, even if the variation in judgments is taken to reflect violations of specific grammatical principles.

The same general issue affects Pullum's other arguments. Take the fact that people are able to give *grammaticality judgments of sentence fragments*. Admittedly, this fact cannot be accounted for by a competence grammar that employs a generative procedure that must start with, or terminate in, an S-symbol. But that is not a problem, as the relevant fact will have to be accounted for in any case by any description of the language faculty at the algorithmic level. It is well known that parsing is incremental (see Gorrell, 1995 and references mentioned there). People are able to recover an interpretation for incoming language word by word. This implies that the performance mechanisms, while designed to operate in accordance with the rules of grammar, must be able to deal with incomplete structures. But if an account is available (or must be developed) for the interpretability (and acceptability) of incomplete structures at the algorithmic level, we are not required to provide an additional account at the computational level (that is, as a part of our competence grammar).

Similarly, there is no problem for a conception of the syntax as a generative procedure arising from the *lexical independence* of grammaticality judgments. Lexical dependence is incorrectly predicted to exist by models of the language faculty in which the grammar is seen as a generative procedure and is held directly responsible for parsing. But there is no need to make this combination of assumptions. The problem dissolves if parser and grammar are taken to be descriptions of the language faculty at different levels. The robustness of the parser makes it likely that the parsing process will not terminate if a terminal is identified whose phonology does not match an existing word. In fact, this must be so in view of the phoneme restoration effect. In case the input contains a phonological form that is not part of a speaker's permanent lexicon, a good strategy might be to store the relevant form in a 'temporary lexicon', and to try and identify a meaning for it (presumably, this is how new words are learned). Again, this can and should be dealt with at the algorithmic level; there is no need to burden the competence grammar with it.

Finally, Pullum points to the existence of *syntactic quandaries*—cases like (11), where there does not seem to be an expression of the form specified that is fully grammatical and has the intended meaning.[3]

(11) $^{?}$Nobody reviewed he/him/his and I's/me's/mine book.

The idea behind the argument is that a generative procedure either produces a well-formed output or 'crashes' and therefore produces no output at all. In (11)

[3] Pullum also discusses acquisition, but it seems to me that this issue is neutral between the two views of competence grammar under consideration.

there is no grammatical way to express the intended content and yet alternatives do not seem fully ungrammatical either. Quandaries of this type, Pullum suggests, are therefore incompatible with a conception of the grammar as employing a generative procedure.

But it seems to me that this argument is really not very different from the argument from gradience of ungrammaticality, and my reaction is therefore not very different either. As long as the grammar is taken to be a characterization of the language faculty at the computational level, it is possible that there are cases in which a given semantic content cannot be expressed by a structure of a particular form. But in view of the robust nature of parsing, it is entirely possible that in performance some of the ungrammatical candidate structures can be associated with the intended interpretation through repair mechanisms of various types, giving rise to an experience in subjects that is neither one of grammaticality nor one of ungrammaticality.

6. Conclusion

None of this means that competence grammar is *not* constraint-based. It just shows that we need better arguments to decide the issue. One kind of argument that I personally find intriguing is frequently used in Optimality Theory. In this constraint-based theory, the evaluation procedure for candidate structures is defined in such a way that a set of constraints used to describe a given language automatically generates a language typology, thus increasing testability. Such typological predictions of course do not follow from any known generative procedure.

Department of Linguistics
University College London

References

Abney, S. 1991: Parsing by chunks. In R. Berwick, S. Abney and C. Tenny (eds), *Principle-Based Parsing: Computation and Psycholinguistics*. Dordrecht: Kluwer Academic, 257–78.

Berwick, R. and Weinberg, A. 1984: *The Grammatical Basis of Linguistic Performance*. Cambridge, MA: MIT Press.

Chomsky, N. 1965: *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

Chomsky, N. 1986: *Barriers*. Cambridge, MA: MIT Press.

Gorrell, P 1995: *Syntax and Parsing*. Cambridge, MA: Cambridge University Press.

Kolb, H-P. and Thiersch, C. 1991: Levels and empty categories in a principles and parameters approach to parsing. In H. Haider and K. Netter (eds), *Representation and Derivation in the Theory of Grammar*. Dordrecht: Kluwer Academic, 251–301.

Marcus, M. 1980: *A Theory of Syntactic Recognition for Natural Language*. Cambridge, MA: MIT Press.

Marr, D. 1982: *Vision*. New York: W.H. Freeman.

Neeleman, A. and van de Koot, H. 2010: Theoretical validity and psychological reality of the grammatical code. In M. Everaert, R. Goedemans, B. Hollebrandse and H. van Riemsdijk (eds), *The Linguistics Enterprise*. Amsterdam: John Benjamins, 183–212.

Wagers, M., Lau, E. F. and Phillips, C. 2009: Agreement attraction in comprehension: representations and processes. *Journal of Memory and Language*, 61, 206–37.

Warren, R. 1970: Perceptual restoration of missing speech sounds. *Science*, 167, 392–93.