

Prolegomena to a Neurocomputational Architecture for Human Grammatical Encoding and Decoding

Gerard Kempen

Published online: 20 July 2013

© Springer Science+Business Media New York 2013

Abstract This study develops a neurocomputational architecture for grammatical processing in language production and language comprehension (grammatical encoding and decoding, respectively). It seeks to answer two questions. First, how is online syntactic structure formation of the complexity required by natural-language grammars possible in a fixed, preexisting neural network without the need for online creation of new connections or associations? Second, is it realistic to assume that the seemingly disparate instantiations of syntactic structure formation in grammatical encoding and grammatical decoding can run on the same neural infrastructure? This issue is prompted by accumulating experimental evidence for the hypothesis that the mechanisms for grammatical decoding overlap with those for grammatical encoding to a considerable extent, thus inviting the hypothesis of a single “grammatical coder.” The paper answers both questions by providing the blueprint for a syntactic structure formation mechanism that is entirely based on prewired circuitry (except for referential processing, which relies on the rapid learning capacity of the hippocampal complex), and can subserve decoding as well as encoding tasks. The model builds on the “Unification Space” model of syntactic parsing developed by Vosse and Kempen (Cognition 75:105–143, 2000; Cognitive Neurodynamics 3:331–346, 2009a). The design includes a neurocomputational mechanism for the treatment of an important class of grammatical movement phenomena.

Keywords Psycholinguistics · Neurocognitive linguistics · Language comprehension · Language production · Grammatical encoding · Grammatical decoding · Parsing · Unification · Sentence processing · Grammatical movement · Competitive structural optimization · Interactive activation and competition · IAC

G. Kempen (✉)
Max Planck Institute for Psycholinguistics,
PO Box 310, 6500 AH Nijmegen, The Netherlands
e-mail: gerard.kempen@mpi.nl

G. Kempen
Cognitive Psychology Unit, Leiden University,
Leiden, The Netherlands

GEEN BOEK ONTVOUWT HET BREIN

No book unfolds the brain

Lucebert (1952)

Introduction

Grammatical processing is part and parcel of two uniquely human communicative skills: language production (where it is called “grammatical encoding”), and language comprehension (where it is called grammatical decoding, or parsing). The cognitive mechanisms in the service of these faculties construct grammatical utterances during language production, and re-construct such utterances during language comprehension. In this paper, I present the design of a neurocomputational architecture for grammatical processing, that takes into account key insights gained in linguistic, psychological, and neurobiological research. It finds its origin in several papers by Vosse and Kempen (2000, 2008, 2009a)—see also Kempen and Vosse (1989, 1994)—, which describe a neurobiologically motivated model of syntactic decoding called “Unification Space.” This model has become a cornerstone of Hagoort’s (2003, 2005, 2007) MUC framework—Memory, Unification, Control—for the interpretation of neurobiological language processing data (see also Baggio and Hagoort 2011).

The new model developed below extends the Unification Space (U-Space for short) in two directions. First, the design opens a perspective on how the U-Space approach to grammatical decoding can be applied to grammatical encoding as well. This capability is prompted by accumulating experimental evidence (summarized in Kempen et al. 2012) for the hypothesis that the mechanisms for grammatical decoding overlap considerably with those for grammatical encoding. This suggests that the human brain may contain a single “grammatical coder” subserving not only decoding but also encoding tasks. Second, the design proposes a

neurocomputational mechanism for the execution of an important case of *grammatical movement*—the phenomenon that, under certain conditions, constituents seem to occupy a position not in the clause they belong to but in a clause higher up the hierarchy.

The computational architecture described below is a blueprint for the neurocognitive processes that enable the emergence, in real-time, of syntactic structures during sentence comprehension and production episodes. By “neurocognitive” (or “brain-style”), I mean that the coding process takes place in a system consisting of a large number of richly interconnected, relatively simple computational units—assemblies of artificial neurons—, and that the shape of the emerging structures is determined by the parallel flow of activation and inhibition between those assemblies, without centralized, supervisory control. Computational architectures meeting these requirements have hardly been investigated in (psycho)linguistics, where it is typically assumed—usually implicitly—, that syntactic structures arise under control by a central agent. This agent has access to a large database of building blocks: lexical items and their selection restrictions, morphemes and morpho-syntactic features, categorial and functional symbols, etc., out of which it assembles syntactic structures in accordance with combinatorial rules retrievable from a second database.

The new design seeks to narrow the gap between psycholinguistic theories of grammatical processing on the one hand, and neurobiological theories of the cerebral infrastructure of grammatical processing on the other. Nonetheless, a wide gulf will remain. The neurons and the neural nets featuring in the blueprint are rather abstract and artificial; and the proposed mappings between modules within the blueprint and (sub)cortical brain areas, or between processes running in the model and neurophysiological or brain imaging effects, are tentative. The goal of the study is to explore the feasibility of a (psycho)linguistically plausible grammatical coding mechanism that meets major neurobiological and neurocomputational demands, as laid out in the next subsections.

Preview The remainder of this Introduction reviews important design criteria to be met by models of grammatical processing that claim plausibility from neurobiological and neurocomputational perspectives. The section titled “[An elementary neural network for grammatical coding](#)” illustrates the basic design of the coding network in terms of a toy grammar. The section “[A grammatical coding network for a more serious grammar](#)” is the *pièce de résistance*: It describes in detail the components needed to process monoclausal sentences. Then, in the section “[Pluriclausal sentences and cross-clausal movement](#)”, the model is somewhat extended in order to be applicable to sentences consisting of more than one clause, and to enable a

treatment of “Wh-extraction”, as illustration of an important class of grammatical movement phenomena. The section “[A neurocognitive architecture of grammatical coding](#)” explores how grammatical coding in the U-Space interacts with other language production and comprehension processes, and what the proposed model entails with respect to its neurobiological infrastructure. The paper concludes with a list of desiderata and open issues.

Neurobiological Design Criteria

The blueprint is founded on the following three experimental-neurobiological considerations.

Grammatical Encoding and Grammatical Decoding Are Subserved by Largely Overlapping Brain Regions

Syntactic encoding and syntactic decoding are often portrayed as independent performance modalities that only share lexicon and grammar. The cognitive *processing* resources underlying these modalities are supposed to be distinct. Kempen (2000), however, proposes an alternative theoretical approach, based on the observation that the two coding modalities operate in a remarkably similar fashion. In language production as well as language comprehension, the syntactic structure for an upcoming sentence is built incrementally, driven by lexical items. The two modalities react similarly to syntactic priming manipulations (see below), and are similarly involved in so-called attraction errors (e.g. missing Subject-Verb agreement in *The helicopter for the flights are safe*; example from Vigliocco and Nicol 1998). Although these and other empirical observations suggest that the same mechanism underlies syntactic structure formation in the two modalities, they do not rule out the possibility that the brain contains two exemplars of this mechanism—one for each modality. However, Kempen et al. (2012) report behavioral experimental results that support the one-exemplar hypothesis. The data were obtained through a novel experimental paradigm called “grammatical multitasking”. The participants had to read (i.e. decode) and to paraphrase (encode) sentences presented in fragments, responding to each input fragment as fast as possible with a fragment of the paraphrase. For instance, the input sentence *The angry/headmaster/complained:/“I/have seen/a nasty cartoon/of/myself/in/the hall”* had to be paraphrased as *The angry/headmaster/complained/that/he/had seen/a nasty cartoon/of/himself/in/the hall*. (Dashes mark the boundaries between fragments.) The switch from direct speech in the input sentence to indirect speech in the output was cued by the symbol “#####” which, in certain trials, was inserted as a separate input fragment before the quoted sentence, and had

to be “paraphrased” as *that*. The main finding was that syntactic predictions with respect to upcoming input that emanate from decoded sentence fragments, were immediately (online) replaced by syntactic expectations emanating from the structure of the corresponding paraphrase fragments. In terms of the example: After having changed the input pronoun *I* to output pronoun *he*, the participants were not at all surprised encountering the reflexive pronoun *himself* in the input sentence they were reading. In trials where the input sentence included the incorrect reflexive pronoun *himself*, no reaction time delay was observed; actually, the grammatically correct input reflexive (*I have ... myself*) yielded longer reaction times than the incorrect reflexive (*I have ... himself*). Apparently, the two modalities have direct access to, and operate upon, the same (i.e. token-identical) syntactic structure. This is possible only if the syntactic encoding and decoding processes command the same, shared syntactic workspace—which presupposes overlap of a crucial processing resource. (For additional behavioral data of similar import, see Tooley and Bock (2011).)

The hypothesis of a modality-neutral grammatical coding mechanism has recently found strong neurobiological confirmation in syntactic priming experiments. The phenomenon of syntactic priming refers to facilitation of syntactic processing in consecutive sentences with similar syntactic structure. For instance, if the first of two sentences (the “prime”) has passive voice, then the second sentence (the “target”) will be processed more readily if also in passive voice. Importantly, there is not only *within*-modality priming (production of the prime facilitating production of the target, and comprehension of the prime facilitating comprehension of the target) but also *between*-modality priming (in particular, from a comprehension to a production task). Brain imaging experiments by Menenti et al. (2011) and Segal et al. (2012) have shown that the magnitude of the priming effect *between* modalities is comparable to that *within* modalities. They measured the effect in terms of repetition suppression of the BOLD response (fMRI) in cortical areas known to subservise syntactic processing: the left inferior frontal gyrus and the left medial temporal gyrus. This finding strongly suggests that the decoding and encoding modalities share their neurobiological infrastructure, at least to a large extent.

Novel Syntactic Structures Arise Through Parallel Search in a Pre-existing Network of Binding Options

The formation of syntactic structures is typically portrayed as a process of composition: establishing new connections (“bindings”) between pre-existing linguistic building blocks (typically, lexical items retrieved from the Mental Lexicon), in agreement with rules of a grammar. This image presupposes that, during the comprehension or production of a sentence, new connections emerge online, as a kind of one-

trial learning. However, this hypothesis makes highly unrealistic demands on the speed of learning (plasticity) in neocortical brain areas—the regions where syntactic structure formation is generally believed to take place (Friederici 2002, 2011, 2012; Hagoort 2005). As argued convincingly by McClelland et al. (1995; see also O’Reilly et al. 2011), learning in the neocortex is slow, requiring many reinstatements of the to-be-learned associations (cf. Hebbian learning). However, this type of learning yields “long-term memories”, which is clearly unrealistic. Indeed, the “verbatim effect” shows that the bindings emerging during grammatical coding are very short-lived—“short-term” rather than long-term memories. Everyday experience indicates that the syntactic form of a novel sentence tends to be forgotten in a matter of minutes, whereas its meaning can be retained for hours without rehearsal. (Experimental studies of the verbatim effect date back at least to Sachs 1967.) If grammatical bindings could emerge online, due to one-trial learning, they would constitute durable memories—as long-lasting as memories for other aspects of comprehended sentences, in particular, sentence meaning.

In a recent fMRI study into the verbatim effect, Poppenk et al. (2008) searched for the brain structures that enable the detection of whether a sentence was old (unaltered since a presentation a few minutes earlier) or novel (altered semantically or syntactically). They identified the Hippocampus (a subcortical area) and the Entorhinal Cortex (mediating between Hippocampus and neocortical regions) as primarily responsible for novelty detection. Crucially, however, novelty detection in these brain areas was restricted to semantic aspects of the sentence. Detection of syntactic alterations was confined to Broca’s area, a neocortical region. The verbatim effect was replicated as well: After a 15-min pause, the participants reacted at chance level (50%) when having to discriminate between “old” and “new.” In contrast, discrimination between sentences carrying new meanings (but unchanged syntactic forms) was still rather good (80–90%). The overall data pattern proceeding from this study confirms that a one-time presentation of a sentence yields a *short-lived neocortical* representation of its syntactic structure.

This might suggest the hypothesis that the formation and storage of grammatical bindings in the course of sentence processing is subserved by the neocortical infrastructure for verbal short-term memory (STM). However, the capacity of verbal STM (“memory span”: the number of numbers, words, letters recalled in correct order after one presentation) typically does not exceed half a dozen of unrelated items (Cowan 2005; Baddeley 2012), not enough to accommodate the number of syntactically independent bindings needed in sentences of average or above-average word length. These considerations prompt the conclusion that, in the language areas of the neocortex of normal adult language users, many grammatical bindings—those that do not involve lexical items—have a

permanent existence. The permanent bindings need not be learned online anymore, but are ready to be activated (recruited) during new grammatical coding episodes. After having been used in a sentence (in ways to be described in detail below), their activation levels drop gradually, giving rise to short-term memory effects. The constellation of “prewired” bindings that is operative in the course of processing a sentence, functions like a scaffolding that enables the language user to remember much longer sentences than allowed by their verbal memory span.

This conception concurs with accumulating behavioral and neuroimaging evidence in support of the idea that the neuronal mechanism underlying STM is not a general-purpose processing resource, ready to support other cognitive functions (problem-solving, sentence processing, imagery), but instead is a component of the mechanisms enabling those other cognitive functions. Verbal STM, in particular, could be an integral part of sentence processing mechanisms (Gupta and MacWhinney 1997; Acheson and MacDonald 2009; for recent behavioral evidence, see Perham et al. 2009; for neuroimaging evidence (fMRI and rTMS), I refer to Acheson et al. 2011).

To anticipate the model developed in the pages below, I posit that grammatically licensed bindings between (nonlexical) syntactic entities are neocortically represented as a network of binding options. The representations must have been acquired in the course of language acquisition (and some may be innately given). During grammatical coding episodes, lexical items transmit activation to entry nodes of the network. Some of these nodes “resonate” if their morpho-syntactic features match those of the input lexical items. From the resonating items, activation spreads to other nodes via prewired connections that represent permissible bindings. Thus, patterns of activation flow emerge that may be depicted as syntactic trees (or, more generally, as a directed acyclic graphs). Clearly, the success of this approach depends crucially on the set of pre-existing connections—which represent the grammar of the language—and on the quality of the decision/selection processes at branching nodes.

Competition Between Incompatible Emerging Syntactic Structures is Resolved by a Domain-General Conflict Resolution Mechanism in Broca’s Area (Brodmann’s Area 44/45)

Syntactic structure formation often leads to activation of incompatible structures, e.g., when a syntactically ambiguous sentence needs to be decoded, or when several paraphrases (alternative constructions) become available during grammatical encoding. In both modalities, this engenders a conflict that has to be solved rapidly. For instance, should the Prepositional Phrase (PP) *with the binoculars* in *The student saw the teacher with the binoculars* be parsed as a modifier

of *the teacher*, or as a modifier of *saw*? And the speaker who is about to describe a transfer-of-possession event, has to decide between a Prepositional Dative (*The teacher showed the binoculars to the student*) and a Double-Object Dative (*The teacher showed the student the binoculars*).

An fMRI study by January et al. (2009) compared the brain’s responses to syntactic ambiguity in a comprehension task, and to a nonsyntactic task requiring the resolution of a cognitive conflict—the Stroop task, which requires fast identification of the font color of a printed word that denotes a different color (e.g., selecting the response “yellow” to the word *blue* printed in a yellow font). The authors found that the two tasks activated the same region (Brodmann’s area 44/45) within the left inferior frontal gyrus. They interpreted this co-location effect as evidence for Broca’s area as hosting a conflict resolution mechanism that is recruited during the execution of syntactic (sentence comprehension) as well as nonsyntactic tasks involving competition between cognitive representations (e.g. Stroop). Novick et al. (2005, 2010) argue the same point and extend its coverage to grammatical encoding tasks as well. Direct neurocognitive evidence for the hypothesis that, during sentence comprehension, Broca’s area is involved in the resolution of conflicts between incompatible representations proceeds from an fMRI study by (Thothathiri et al. 2012). They compared BOLD responses in participants who read grammatically well-formed sentences of four different types:

- | | |
|----------------------------|--|
| A. Active: | <i>The father was calming the cranky girl</i> |
| B. No-conflict
Passive: | <i>The celebrity was interviewed by a reporter</i> |
| C. Neutral Passive: | <i>The patient was interviewed by the attractive man</i> |
| D. Conflict Passive: | <i>The journalist was interviewed by the undergraduate</i> |

The three types of passives vary with respect to the degree of compatibility between the syntactically and the semantically induced interpretation. Sentence D elicits a conflict because journalists are more likely to be interviewers than interviewees. On the other hand, celebrities are typical interviewees (type B); and type C is neutral in this respect. BOLD responses in various regions of interest within Broca’s area matched with level of conflict (Conflict > Neutral > No Conflict). There was no tendency for passive sentences to elicit stronger BOLD responses than actives. Thothathiri et al. account for this data pattern as follows: In the Active and the No-Conflict passive sentences the syntactically and semantically prompted interpretations converge whereas these interpretations diverge in the Conflict Passives.

These and related neuroimaging data support competition-based computational-psycholinguistic models for important aspects of grammatical decoding (Stevenson 1993; Kempen and Vosse 1994; Vosse and Kempen 2000, 2009a, b; Alday

et al. 2013), and grammatical encoding (Eberhard et al. 2005; Nozari et al. 2011).

Related Neurocomputational Models

Neurobiologically inspired computational models of grammatical decoding (“connectionist parsers”) have been around since almost 30 years. They come in various flavors, and a detailed survey is beyond the scope of this article. So, only the most prominent models can receive mention. A crucial event, at the end of the 1980s, was the demonstration that Recurrent Neural Nets (RNNs) are remarkably good at learning to detect regularities in sequential input strings (typically with backpropagation as learning paradigm). Prime examples are Simple Recurrent Nets (SRN; Elman 1991a, b) and Recursive Auto-associative Memory (RAAM; Pollack 1990). Trained on a corpus of sentences generated by a relatively simple phrase-structure grammar, SRNs were shown to develop “complex distributed representations which encode the relevant grammatical relations and hierarchical constituent structure (Elman 1991a:195). Recently, a new type of RNNs has been designed based on reservoir computing (Hinault and Dominey 2013). Their model could not only learn to parse sentences correctly but it also assigned thematic roles to grammatical constituents; in so doing it could take a previous sentence into account (an elementary discourse processing skill); and it simulated a P600 effect. During the training epochs, the RNNs tend to develop distributed representations of the lexical, syntactic and semantic items populating the grammar to be acquired.

Another approach, also requiring training and typically yielding distributed representations, is based on tensor product representations of filler/role bindings (Smolensky 1990). Recent implementations of this idea are Vector Symbolic Architectures (Gayler 2003), and the connectionist minimalist parser developed by Gerth and beim Graben (2009; see also beim Graben et al. 2008). Kanerva (2009) presents a review of these and other applications of hyperdimensional (neuro)computing.

In spite of these successes, models with inbuilt localist representations of grammar and vocabulary have not lost their appeal. Some localist models served to explore the possibilities afforded by synchronously firing neurons as representations of filler/role bindings (Henderson 1994; Shastri and Ajjanagadde 1993; Hummel and Holyoak 2003). Other authors opt for a localist-connectionist approach because they impute principled limitations to distributed sentence representations (van der Velde and de Kamps 2006). The latter publication also includes debates on the pros and cons of distributed representations, and on representations using synchronously firing neuron assemblies. The localist-connectionist models proposed by van der Velde & de Kamps and by Hummel & Holyoak were recently

criticized by Stewart and Eliasmith (2012) for requiring an unduly large number of neurons if scaled up to realistically-sized vocabularies and grammars.

The conclusion must be that no single paradigm is emerging as the most promising one. Also remarkable is the fact that little attention is paid to simulation of neurobiological data (neuro-imaging results, ERP effects, brain localization)—Dominey’s work seems to be the sole exception—or to psycholinguistic phenomena. And a general issue—not restricted to localist models—concerns the extent to which the models can be scaled up. So, what is the best way to go from here?

In his monograph *The algebraic mind*, Marcus (2001, Ch. 4) critically examines connectionists approaches to simulating an essential feature of high-level human cognition (including language): the representation and manipulation of recursively structured knowledge. After pointing out serious shortcomings of several neural coding schemes, including RNNs, he proposes an alternative neural encoding scheme for hierarchical knowledge structures, and concludes as follows:

For a variety of reasons, admittedly none decisive, I have argued in favor of a representational system that consists of a set of empty templates (treelets) that contain banks of distributed nodes (register sets) that contain encodings for primitives. Such a system does not require an undue number of nodes or connections or a system that rapidly constructs new connections between nodes, and yet it is able to encode the range of structures expressible in complex recursive systems such as human language.” (2001:117).

A decade later, Marcus (2009, 2013) still subscribes to this general approach but expresses doubts about whether

“the mind has a neurally realized way of representing ‘arbitrary trees,’ such as the syntactic trees commonly found in linguistics. [...] [We may] be forced to rely on a sort of cobbled-together substitute for trees, in which linguistic structure can only be represented in approximate fashion, by mean of sets of subtrees (“treelets”) that are bound together in transitory and incomplete fashion. [...] Our brains may thus be able to afford some sort of approximate reconstruction but not with the degree of reliability and precision that veridically represented trees would demand. [...] [A] host of facts, ranging from the human difficulty with parsing center-embedding sentences to our difficulties in remembering verbatim syntactic structure, are consistent with this notion.” (Marcus 2009:163).

The SINUS model by Vosse and Kempen (2009a) fits this description pretty well (“sets of subtrees (“treelets”) that are bound together in transitory and incomplete fashion”), thus enabling it to simulate a range of psycholinguistic facts that suggest the human mind is only “able to afford some sort of

approximate reconstruction” of veridical syntactic trees. This approach will be continued in the present paper.

The next section lays out some core ideas of the new U-Space model of grammatical encoding and decoding in terms of a toy example.

An Elementary Neural Network for Grammatical Coding

A Simple Network for a Toy Grammar

Consider the rewrite rules in (1), which together generate strings like *dogs bite*, *cats fight*, *cats dogs bite fight*, and *mice cats dogs fight kick bite*, including the famous *dogs dogs dogs bite bite bite*. The plus sign indicates obligatory word order, and optional constituents are parenthesized.

- (1) $S \rightarrow NP+V$
 $NP \rightarrow N (+ S)$
 $N \rightarrow \textit{dogs, cats, mice}$
 $V \rightarrow \textit{bite, fight, kick}$

In the network representation of this grammar in Fig. 1, the rules appear as treelets whose branches connect nodes named after the lexical (N, V) and phrasal (NP, S) rule symbols. The figure shows two exemplars of every treelet, sufficient to process the sentence *Dogs cats bite fight*; sentences with additional levels of recursion need more exemplars. The rule symbols are represented by artificial (abstract) neurons, the branches by connections between such neurons. The neurons are processing units capable of executing simple numerical computations in response to local signals coming in via input wires. The signals are activation values—positive (excitatory) or negative (inhibitory). The computation within a neuron yields an output signal called the neuron’s *activation level*. If this level exceeds a threshold, the node “fires”; that is, it relays its activation to all nodes at the other end of its output wires. Weights associated with these connections can modulate the amounts of activation or inhibition conveyed; however, unless indicated otherwise, all weights are equal to 1.

An additional factor influencing a neuron’s activation level is decay—a spontaneous, gradual reduction of the activation level taking place independently of input or output. Decay is modeled as a negatively accelerated function of time (decay is strong initially but flattens out slowly). All connections are bidirectional, implemented in the form of two parallel unidirectional connections where activation travels in opposite directions (and whose weights need not be identical). The bidirectionality enables reverberating circuits that temporarily reduce the effect of decay: sustained activation. (Unidirectional connections can be modeled as bidirectional ones, with one of the weights set to zero.)

Inhibition links have the effect of reducing the amount of activation transmitted over the inhibited connection. For instance, consider the following network containing neurons A, B, and C: A receives activation from B and C, both trying to bind A; and there is an inhibitory link between connections B–A and C–A. When activated, neurons B and C send roughly equal amounts of activation to A. If neuron B is the first to start emitting activation, the B–A connection also starts sending inhibition to C–A.¹ Subsequently, if neuron C is activated and sends activation to A, the amount of activation flowing via C–A will be systematically reduced with the amount of inhibition coming in from the B–A connection, thus rendering C–A the “weaker” connection. The model includes “winner-take-all” circuits that amplify differences in the amount of activation passing through competing connections, until a difference threshold is reached and the weaker connection is knocked out.

Treelet exemplars of the same treelet type are activated (“recruited”) one-by-one by a *router* circuit.² In the example of Fig. 1, an S-router recruits the left-hand S-treelet exemplar first, then the right-hand one; likewise, an NP-router recruits the two NP-treelet exemplars from left to right. Once recruited, a treelet is able to bind another treelet (thus becoming the latter’s daughter), and to be bound by another treelet (as the latter’s parent) or by a lexical item (which becomes the “anchor” or “head” of the treelet). If a recruited treelet does not receive any binding and thus remains isolated, its activation decays quickly because there is no reverberating circuit sustaining it.³

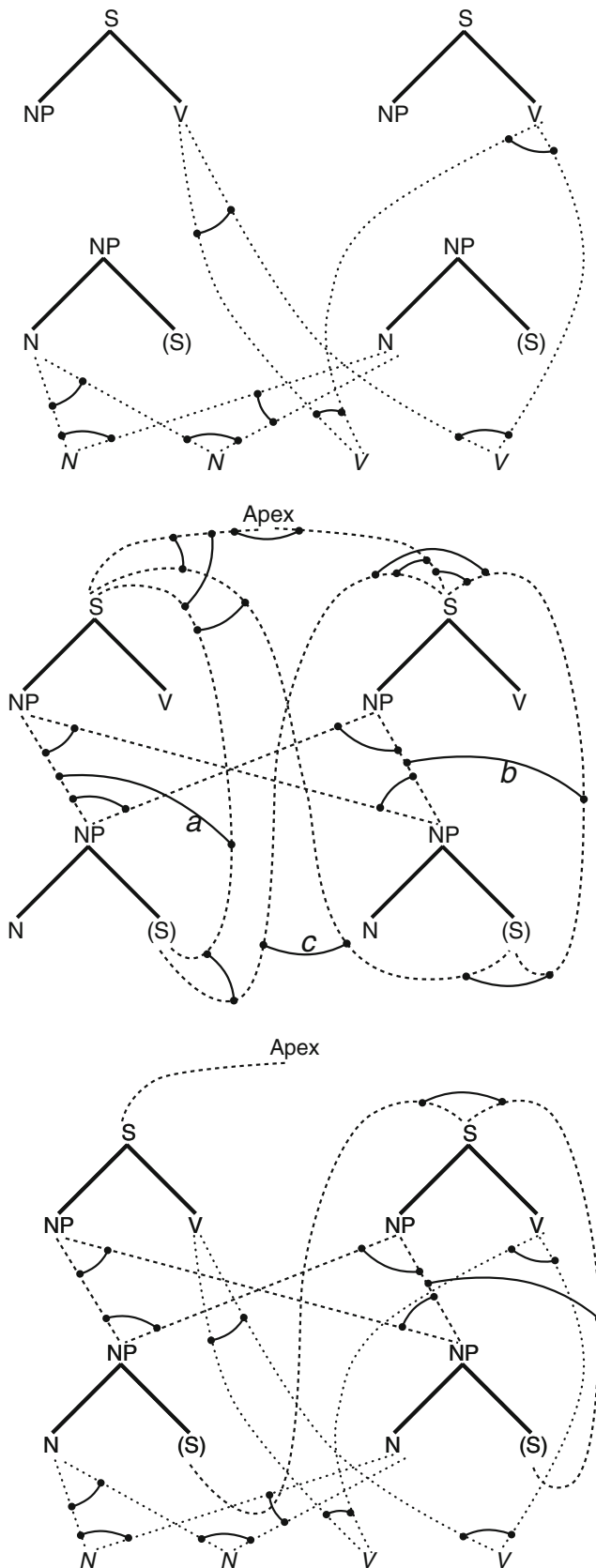
The process of decoding the example string starts when the first input word, *dogs*, sends activation to the NP-treelets (Fig. 2a). Only the left NP-treelet can respond (“resonate”) because this is the one currently being recruited by the NP-router. Because the left S-treelet also happens to be active (due to the activation from the S-router and from the Apex node; see lower panel of Fig. 1), the NP foot node of the left S-treelet responds, meaning that NP *dogs* is now bound to the left S-treelet. Both treelets start sending activation to other—as yet dormant—nodes, as indicated by the NP-to-NP and the (S)-to-S links in Fig. 2a. These links represent *predictions* of possible continuations of the string.

Next, the NP- and S-routers recruit the two right-hand treelets in the figure. The noun *cats* then binds the NP-treelet,

¹ Actually, this description is a simplification because connections send activation to other *neurons*, not to other connections.

² For a recent computational routing model embodying a high level of neurobiological plausibility, see Zylberberg et al. (2010). The task this model was designed to execute is very different from the task intended here. Nevertheless, I mention it here because it succeeds in modeling the execution of novel sequential tasks on massively parallel neural hardware.

³ For recent fMRI evidence in support of the hypothesis that activation generated in response to incoming words is maintained for the duration of a syntactically coherent, and hence parsable constituent, see Pallier et al. (2011).



◀ **Fig. 1** Network version of the toy grammar in (1). *Upper panel.* The treelets (straight continuous lines) correspond to the first two rules in (1); two exemplars of each rule are shown. S=Sentence; NP=Noun Phrase; N=Noun; V=Verb. Dotted lines: excitatory connections between lexical categories mentioned in (1) and corresponding foot nodes of the treelets. Continuous curved lines ending in two black disks: inhibitory connections between incompatible bindings. *Middle panel.* Each dashed line connects the root node of one treelet with a foot node of another treelet. Inhibitory connections run between incompatible bindings. The links labeled *a*, *b*, and *c* serve to prevent loops: They force the system to choose one of the two possible dominance relations between treelets. For instance, if the top left S-treelet dominates the bottom left NP-treelet, the latter cannot simultaneously dominate the former. Two additional links of this type exist but have not been drawn as they play no role in the example. The Apex is a special node that immediately dominates the root nodes of all S-treelets. *Lower panel.* The connections of the top and middle panels are combined here, except for two dashed lines that, in the middle panel, start at the top left S-node. As indicated in the text, this S-node is the first to be recruited by the S-router, and will become the root of the entire emerging sentence (unless changing inhibitory forces compel a reversal of the dominance relations during the decoding episode). The excitatory connection between this S-root and the Apex inhibits all other links leaving this S-node, as well as all links from the Apex to other S-treelets

which in turn binds the NP-foot of the right-hand S-treelet (Fig. 2b). This NP-foot cannot be bound by NP *dogs* because the corresponding NP-to-NP connection receives inhibition from the already established NP-to-NP connection in the left part of the figure. The fact that the second S-treelet has now been bound means verification of the prediction that the first noun could host an embedded clause (S-treelet). The (S)-to-S link in Fig. 2c, which is not hindered by any competition, realizes the embedding.

Finally, the network has to accommodate the two verbs. The first one (*bite*) is confronted with binding options in both S-treelets. Of these, it selects the most recently recruited one (see Fig. 2d, which also depicts the final parse tree). The reason for this choice is twofold. First, the activation running in this treelet has undergone less decay than the other, “older” one, granting its binders a higher likelihood to emerge as winner in a competition (“recency effect”). Second, the system checks the word order that a binding implies, against the actual input word order. (The mechanism taking care of this will be explained in section “U-Space input and output”.) In the next subsection, I turn to the determination of word order, including the binding choice of the second verb (*fight*).⁴

⁴ Conceptual and referential factors can also help to prevent incorrect bindings (see subsections on these topics below) by sending excitatory or inhibitory signals to tentative binding partners depending on the (im)plausibility of a reading entailed by the binding. However, such a mechanism would not help to prevent the interchange of *bite* and *fight* (nor, that matter, of *dogs* and *cats*); it may even increase the probability of exchanges, as in *man bites dog*.

Fig. 2 Decoding example string *Dogs cats bite fight*.

a Activation flow in response to first input word. The tree at the right-hand side shows the resulting parse tree in conventional linguistic notation. The dotted line in the parse trees corresponds to the bolded dotted line in the network diagram. The Apex node is not shown. **b** First part of the activity elicited by the arrival of the noun *cats*. The S-to-(S) link in the right-hand part of the network is knocked out by a loop-preventing inhibitory link (link *b*). **c** Final part of the activity elicited by the arrival of the noun *cats*. **d** Upper panel. The verb *bite* binds the second S-treelet. Lower panel. The finally resulting parse

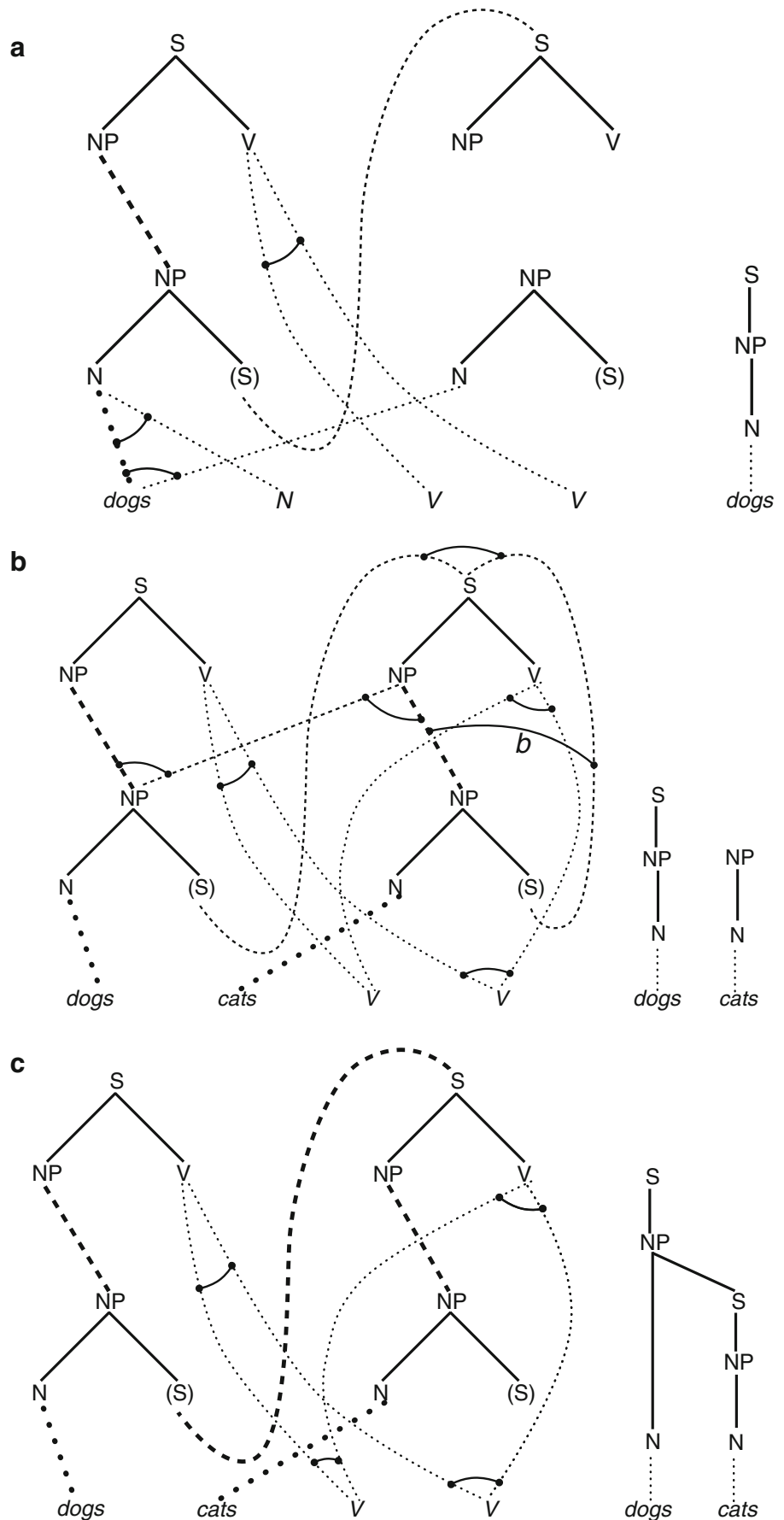
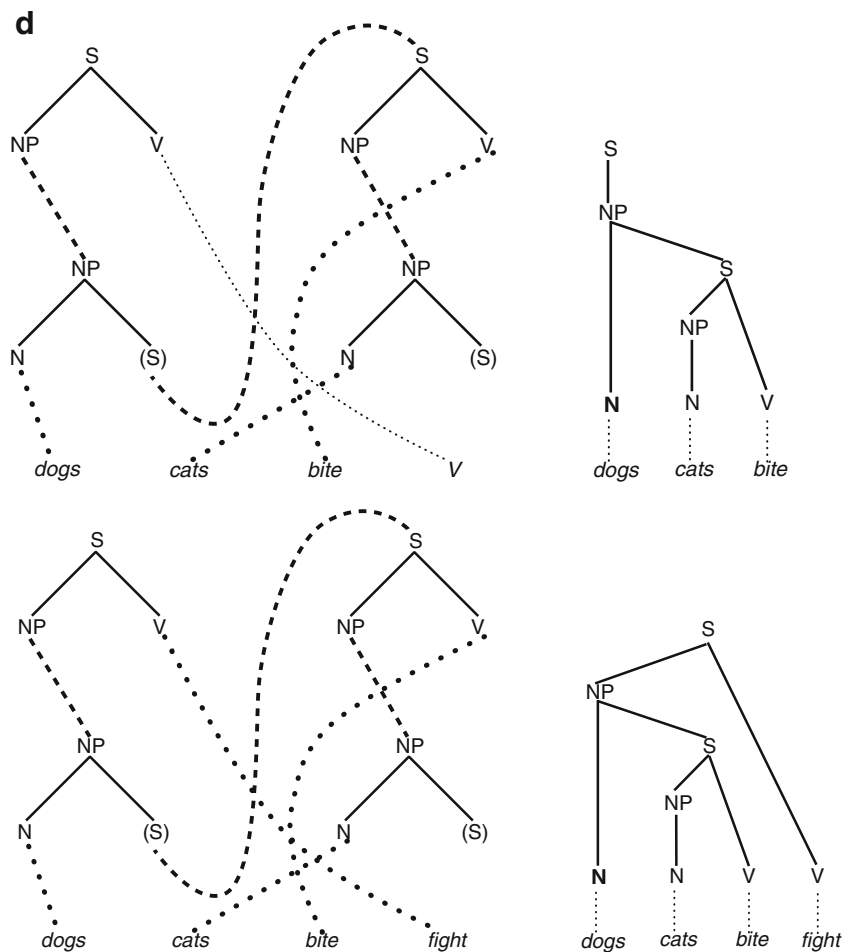


Fig. 2 (continued)



The Treatment of Word Order

Many modern grammar formalisms decompose rules like $S \rightarrow NP + V$ into a hierarchical part that expresses the parent–child relations between higher and lower constituents, and a linear-order part expressing the precedence relations between child constituents. In line with these proposals, I assume that the root and foot nodes of a treelet are not connected directly, but indirectly via a *Recursive Transition Network* (RTN) associated with the root node. RTNs belong to the class of Augmented Transition Networks developed by Woods (1970) as natural-language parsers; see also Kaplan (1972).⁵ The states of RTNs proposed in the present network model correspond to positions (slots) in a one-dimensional array called *topology*. Different treelet types have different topology types.

As described in the previous subsection and illustrated in Fig. 2, the activation pattern emerging bottom-up in response

to the sequence of input words represents *binding options*—between lexical items and anchor nodes of treelets, or between the root of one treelet and a non-anchor foot of another treelet. The RTNs carry out a twofold function. The first one is to exert top-down control on the order in which to-be-bound nodes resonate to the prospective binder node(s). As will be illustrated next, this ensures that the emerging structure is a tree (in the standard linguistic sense of a rooted acyclic graph with branches ordered from left to right). The second function of the RTNs is enabling sequential readout of the terminal nodes of a tree during grammatical encoding (see subsection “U-Space Input and Output”).

Figure 3 depicts the RTNs at work during the process of decoding the sample string of Fig. 2. As soon as the upper left S-treelet is recruited, it starts up its RTN (transition #2), which then attempts to fill the position S_{pl} . As the wiring is such that depth-first transitions are tried first, the RTN sends activation to the NP-foot via edge #4. This activation is one input to an AND-gate (not shown in the figure) that fires if two input wires are active simultaneously. This condition is fulfilled if the foot node is resonating to bottom-up activation coming in from the root of the lower left NP-treelet; that is, if

⁵ RTNs are automata capable of parsing context-free languages. However, natural languages belong to the class of mildly context-sensitive languages (Joshi 1985), due to grammatical movement (see section “Pluriclausal sentences and cross-clausal movement”).

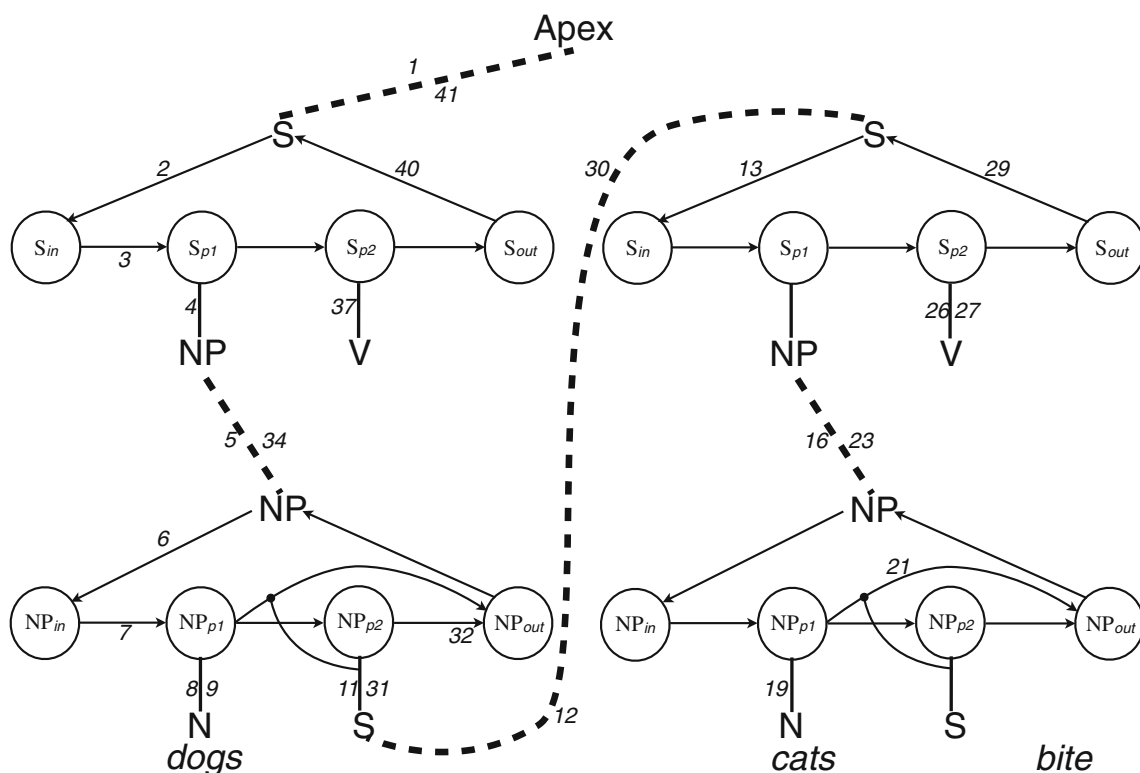


Fig. 3 RTNs in the treelets of Figs. 1 and 2. The snapshot shows the situation after the first two words of the string *Dogs cats bite fight* have been processed and the verb *bite* is entered. Circles denote RTN states, which correspond to linear (“topological”) positions. S_{p1} means “first position in S-treelet”; NP_{p2} means “second position in NP-treelet”, etc. In

the course of the parsing process, the RTNs get linked, giving rise to a context-free transition network that is traversed in a depth-first, left-to-right manner. In the example, the transition network is traversed in 41 steps (see the numbered edges), starting and finishing at the Apex. (Only some of the edge labels are mentioned.)

a binding has been established (possibly after a competition with other binding candidates). In the present case, the binder is an NP treelet that has been recruited by the NP router and is not inhibited by any competitor. The AND-gate now relays activation to the root node of the binder, thereby passing control to the latter’s RTN. Meanwhile, the resonance representing the binding at the end of edge #4 goes on for a while, trying to inhibit and ward off binding options launched at later points in time from elsewhere in the network, until succumbing to decay. Transitions #5 through #8 can now be traversed for free. The NP-anchor node accepts *dogs* as binder, and RTN control moves on to state NP_{p1} via edge #9. From here, two transitions are attempted in parallel, one of them checking the prediction of an embedded clause (#11). Recruitment of the right-hand S- and NP-treelets, and arrival of the noun *cat* enable edges #12 through #19 to be traversed, and the latter noun to bind the N-foot. Next, edge #21 is followed because the competing transition to NP_{p2} , which is tried first, fails to return an embedded clause and decays, thereby disinhibiting the direct link from NP_{p1} to NP_{out} .

How does the RTN find its way back to the Apex? The answer presupposes that every treelet exemplar is embedded in a fixed network with prewired links to all (and only) the treelet exemplars that are eligible as binding partners. Therefore,

descending upon the root of every treelet, there is a limited number of foot-to-root connections, each representing a potential binding of that treelet. This suggests an implementation where every such descending link is paired with an ascending link, and where the latter link is then selected as the return path for the RTN on its way to the Apex. Presumed here is that the resonance in a descending link lasts long enough to enable the root node to select the correct ascending link. A neuronal mechanism as sketched here can fulfill the function of the *pushdown memory storage* (“stack”) in hierarchical procedure calls in digital computers (see Pulvermüller 2010, for a related approach).

Continuing now the treelet traversal, the verb *bite* attaches to the right-hand S-treelet due to the fact that the V-node at the end of transition #26 accepts the verb as binder. (The activation that is running now between the resonating V-foot (edge #26) and the lexical item *bite*, easily knocks out the competing link between *bite* and the V-foot of the left S-treelet because the resonance at this V-foot is decaying.) RTN traversal continues via edges #27 through #37. The final verb *fight* is accepted as binder of the V-node at the end of edge #37. (Activation running in the corresponding connection extinguishes the competing link between *fight*

and the V-node at the end of edge #26.) Transitions #38 through #41 return control to the Apex. The resulting parse tree is depicted in the lower right part of Fig. 2d.⁶

Completion of an RTN traversal does not necessarily mean the end of the decoding episode. Among the continuation possibilities are the following. First, the activation in the current network extinguishes gradually due to decay, so that the treelets can be recruited again for a novel sentence (or further downstream in the same—presumably long—sentence). Second, the system can be reset instantaneously in response to explicit end-of-sentence signals (intonation contour, printed symbols). A third possibility is that more input words are added to the current sentence. In that case, the Apex will initiate an additional RTN traversal trying to accommodate the new input words. This may change the balance of power in the existing network due to activation and inhibition originating from newly recruited treelets and new bindings. This may give rise to “garden-path phenomena” (if the new words entail that the system has been “led up the garden path” and a chosen RTN trajectory needs to be abandoned and replaced; see footnote 8 for an example). The fact that these and other phenomena related to syntactic and lexical ambiguity are run-of-the-mill entails that many sentence decoding episodes require iterative RTN traversal.

Grammatical Encoding in the U-Space

Grammatical encoding is driven by conceptual input to the U-Space, via the same bidirectional connections that subserve the conceptual interpretation of parse trees during grammatical decoding. In the toy example, no conceptual input has been defined. However, let us assume *ad hoc* that the following to-be-encoded toy conceptual representation underlies the example sentence *Dogs cats bite fight*: Some DOGS are “patients” of BITING, some CATS are “actors” of this BITING, and the DOGS are actors of FIGHTING. The capitalized concepts are “lexicalized” as the nouns and the verbs of the toy grammar; and the to-be-expressed conceptual relations map onto U-Space treelets as combinatorial constraints:

⁶ When arriving at a foot node, the RTN only checks whether there is a resonating (winning) binder, and if so, proceeds to the next transition. However, the bottom-up competition from which the binder emerges, is entirely at the mercy of local forces (activation and inhibition strengths) and local information (the feature matrices of prospective unification/binding partners). This leaves open the possibility of exchanges: Two binders with similar grammatical properties happen to get swapped, and the RTN has no reason to mistrust either. Hence, the system needs a check that compares RTN-implied word order against actual input word order, and prevents the RTN from moving to the next position upon detection of a mismatch (see Subsection “How does the U-Space communicate its output to other processing modules?”).

Actors should bind the same S-treelet as the action they execute; and patients undergoing an action should bind the same NP-treelet as the clause expressing that action. The treelets are supposed to contain circuitry enabling them to access and verify these constraints. If a tentative binding cannot be verified, the treelet inhibits the corresponding connection, which in consequence will have great difficulty defeating its competitors.

The to-be-encoded conceptual structure can be entered into the system incrementally, in increments of any size, and in any temporal order. (This level of freedom mirrors a basic property of human sentence production.) Fig. 4 shows the course of events elicited by four somewhat arbitrarily chosen increments: (1) DOGS, (2) DOGS actors of FIGHTING, (3) CATS actors of BITING, and (4) DOGS patient of (3).

The grammatical encoding process starts with recruitment of both left-hand treelets, and the noun *dogs* trying to unify with the NP-treelet by sending activation to the N-foot. This succeeds after the RTN has passed through transitions #1 through #8. As the verb *fight* cannot open an embedded clause at position NP_{p2}, this transition decays and disinhibits #10, which allows the verb to bind the V-foot at the end of edge #15. The S-treelet does not inhibit this binding since it conforms to the combinatorial constraint deriving from the second increment. From here, edges #16 through #19 can be crossed for free, returning control to the Apex with a well-formed string *dogs fight*. The Apex initiates a second RTN traversal; the right-hand treelets that the S- and the NP-router have recruited in the meantime, provide binding sites for additional words. This time, the left-hand RTNs consist of “beaten tracks” #1 through #9. Meanwhile, responding to the third increment, the noun *cats* and the verb *bite* have begun to activate the N- and V-foot nodes in the right-hand treelets. In conjunction with verification of the constraint deriving from the fourth increment, this means that an embedded clause materializes in position NP_{p2} of the left-hand NP-treelet, and edge #10 will be bypassed. The remainder of the encoding process follows the same trajectory as the decoding process depicted in Fig. 3.

Missing yet from this illustration of grammatical encoding is another duty of the RTNs: *readout*, i.e., the actual delivery of an output string of lexical items. More on this follows in Subsection “Input and output”.

A crucial insight to be gained from this section is that grammatical encoding and grammatical decoding can indeed be handled by one and the same syntactic processing system, and that the operations in this system need not reverse direction when switching between encoding and decoding modality—although, at first impression, the two modalities of grammatical coding operate in rather different manners and in opposite directions. The main difference between the

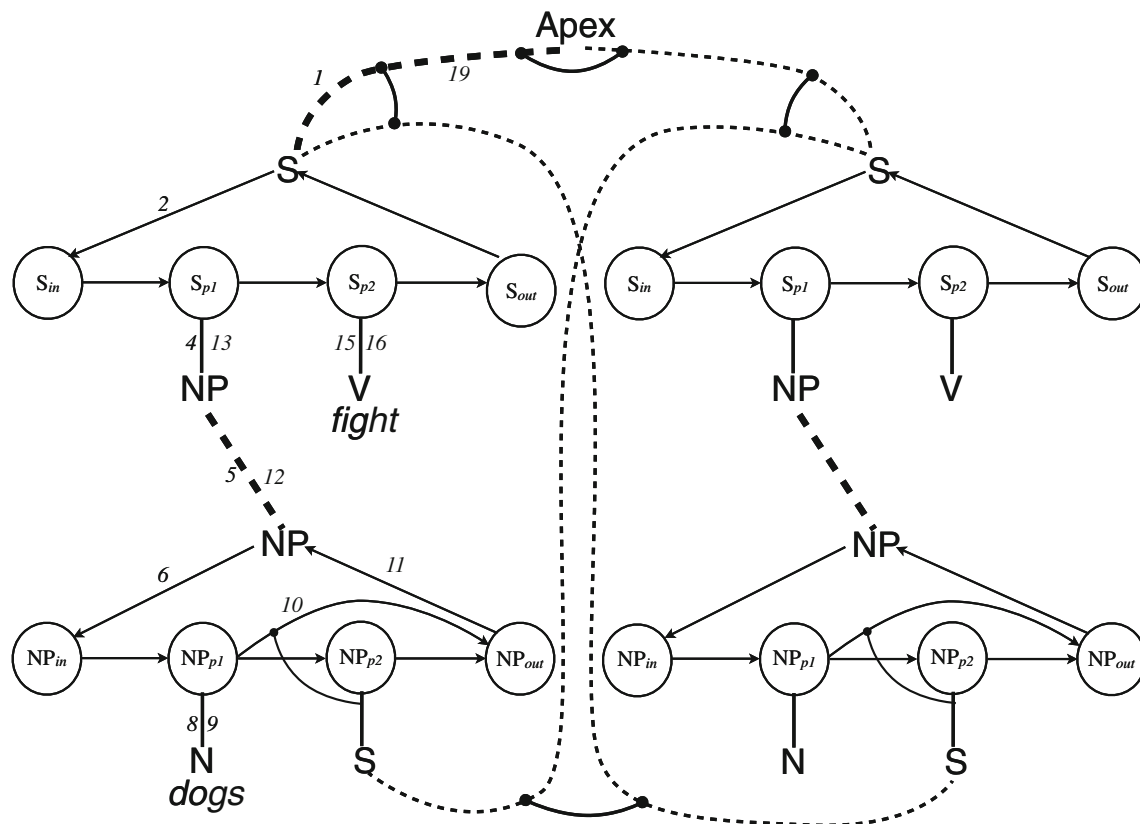


Fig. 4 RTN traversal during grammatical encoding of sample string *Dogs cats bite fight*. The snapshot shows the situation after processing the first two conceptual increments: DOGS, and DOGS actors of FIGHTING

two coding modalities resides in the way they treat input: During encoding, the conceptual structure is “clamped” (i.e., not allowed to change except for the addition of increments) and enabled to drive the flow of excitation and inhibition in U-Space; during decoding, it is the sequence of input words that undergoes clamping.

A Grammatical Coding Network for a More Serious Grammar

In this section, I apply the mechanism developed above to a small fragment of Dutch, my native language. The notational and graphical conventions serving to characterize the syntactic structures originate for the most part from the psycholinguistically motivated grammar formalism of Performance Grammar (PG; see Kempen and Harbusch 2002, 2003; and Harbusch and Kempen 2002).

Syntagmas

The Unification Space does not only process treelets for clauses (S-type) and Noun Phrases (NP-type) but also for Prepositional Phrases (PP), Adjectival/adverbial Phrases

(AP), Determiner Phrases (DP), Conjunction Phrases (CP), and Cardinal Number Phrases (CNP). From now on, I will use the term *syntagma* to refer to syntactic treelets. Syntagmas have a somewhat different structure than the treelets figuring in the previous section. Each syntagma exemplar can be *bound* by one *lemma*, called its anchor or Head.⁷ Lemmas are word-sized lexical items that specify the morphological and syntactic properties of words. These properties include word class (part of speech, PoS); gender, number and case of nouns and articles; transitivity of verbs; and several other properties. Lemmas should be distinguished from *lexemes*, which specify the phonological (sound) form of words, and from *concepts*, which specify word meanings. The lemma that binds a syntagma of a certain type should belong to a specific

⁷ Note that, while each treelet *includes* a lemma, each syntagma can be *bound* by a lemma (in fact, by a lemma of the correct word category). This means that, strictly speaking, the model developed here is not based on a “lexicalized” grammar—for the simple reason that the tree-like structures are not retrieved from the Mental Lexicon (but activated in U-Space in response to lemma features). This difference is in agreement with recent fMRI data showing that, as far as the linear order and the morpho-syntactic properties of input items are concerned, the grammatical decoding process treats Jabberwocky-type sentences and meaningful sentences similarly (Pallier et al. 2011). The nonsense words in Jabberwocky sentences cannot be assumed to be represented as lexical entries, let alone as treelets (*o.c.*:2526).

PoS or to one of a few specific PoSs. For instance, NP-syntagmas can be anchored/headed by nouns or pronouns, clauses by main verbs, auxiliary verbs, or modal verbs. The non-head foot nodes of syntagmas specify a combination of a grammatical function (e.g., Subject, Direct Object, Indirect object, Modifier, Complement) and a phrase/constituent (S, NP, AP). Examples are NP functioning as Subject (SUBJ-NP), AP in the function/role of Modifier (MOD-AP), and subordinate clauses functioning as Complement (CMP-S) or as Modifier (MOD-S; called adverbial clause in school grammars).

Syntagmas are generic phrasal templates, in the sense that each type can yield phrases of many different shapes. Binding a syntagma to a lemma causes specialization of the template. For instance, when an S-syntagma is bound by an intransitive verb lemma, activation of Direct and Indirect Object nodes remains blocked, so that no NP-syntagma can unify with them; and if an NP-syntagma is bound by a relative pronoun, it can no longer host a determiner, modifier, quantifier, etc. Syntagmas include RTNs that specify the possible linear positions of anchoring and non-anchoring foot nodes. Another kind of neural nodes within syntagmas controls agreement between foot nodes (e.g., number and person agreement between the Subject-NP and the Head-Verb of an S-syntagma; and gender and number agreement between Article (determiner) and Head of an NP).

The U-Space contains several exemplars of every type of syntagma. I will not provide detailed estimates of the number of exemplars of every type needed for coding shorter and longer sentences. However, five to ten exemplars of every type is likely to suffice. In case of long sentences that approach the upper limit of the human grammatical processing capacity, we may assume that syntagmas recruited early on in the sentence, gradually fall victim to decay—or are actively reset—and can be re-used further downstream.

Figure 5 illustrates the make-up of an S-syntagma that could drive the encoding or decoding of a monoclausal sentence. The upper half shows a row of topological nodes and, at the top, a node representing syntactic properties of the clause *per se* (main, subordinate, relative, complement, etc.). The node sequence between S_{in} and S_{out} forms an RTN that, traversed from left to right, yields a sequence of constituents comprising a clause. The node labels (F1, M1, ..., M6, ..., E2) derive from the English translation of the original German names *Vorfeld* (Forefield), *Mittelfeld* (Midfield), and *Nachfeld* (Endfield). The lower row of nodes are foot nodes labeled with the type (NP, AP, S, etc.) and function (Subject, Direct Object, Head, etc.) of potential binding syntagmas or lemmas. The links connecting the two rows of nodes indicate optional positions of binders in the topology. These links and the topologies are simplifications of the actual word order rules. For instance, in main clauses of Dutch, the Head verb always goes to position M1 (“verb-

second”), and in subordinate clauses of any type it always goes to M6 (“verb-final”).

Word order in syntagmas representing other types of constituents is treated in similar manners. Figure 6 depicts the syntagma for Dutch NPs.

The system’s behavior depends in considerable measure on the temporal coordination between the bottom-up and the top-down aspects of structure formation. Clearly, the bottom-up establishment of binding relations must precede top-down acceptance (through a check whether there is a clear winner) and readout by the RTN. But, how closely should the RTN “shadow” the binding process? Allowing large intervals between the emergence of a binding and its acceptance by the RTN may cause memory problems due to decaying activation. On the other hand, if the RTN shadows emerging bindings very closely, it runs the risk of accepting and reading out momentary bindings that are soon undone under the influence of subsequent lexical input.⁸ I will assume that the RTN shadows input at a distance of just a few lemmas, thus allowing a short look-ahead interval.

Given that some constituents have multiple placement options, the RTN should include circuitry preventing a constituent to be placed and read out more than once. The solution I propose uses binary features, called “*place tags*”, belonging to the feature collection associated with root nodes of syntagmas. When a syntagma is recruited, the place tag feature is activated, indicating that the syntagma is seeking placement in a topology. Once the RTN has read out a constituent, it *deactivates* the place tag. Deactivated place tags block any further readout but, importantly, they do not block relocation of a read out constituent to another topology. Such a relocation will remain covert (tacit, invisible) because no readout will take place at the new location. In the section

⁸ An example from SINUS (Vosse and Kempen 2009a, p. 342) illustrates how new lexical input may help the decoder to escape from a blockade caused by a local lexical/syntactic ambiguity. Sentence (ia) (adapted from Ferreira and Clifton 1986) is initially parsed as a main clause with *examined* as past-tense finite Head verb and *the lawyer* as Subject, as in (ib) where this analysis is the correct one. However, in the Mental Lexicon, the verb form *examined* is associated not only with the lemma of an active verb, but also with that of a passive past-participle, which includes an optional “agentive” Modifier foot node bound by a PP with *by* as Head. These lemmas inhibit one another, and this inhibition transfers to the S-syntagmas recruited by them in the Unification Space. The arrival of *by* boosts the activation the passive lemma in the Mental Lexicon, and indirectly that of the agentive Modifier foot node of the passive S-syntagma in the U-Space. Simulation runs with SINUS show that the original parse gives in to one where *examined by the court* plays the role of reduced relative clause attached to *lawyer*, provided that the initial activation of the passive participle is not too low. The reanalysis requires that the original binding of *lawyer* to the Subject foot node of the S-syntagma anchored to the active verb form is undone.

(ia) The lawyer examined by the court was found guilty

(ib) The lawyer examined the evidence

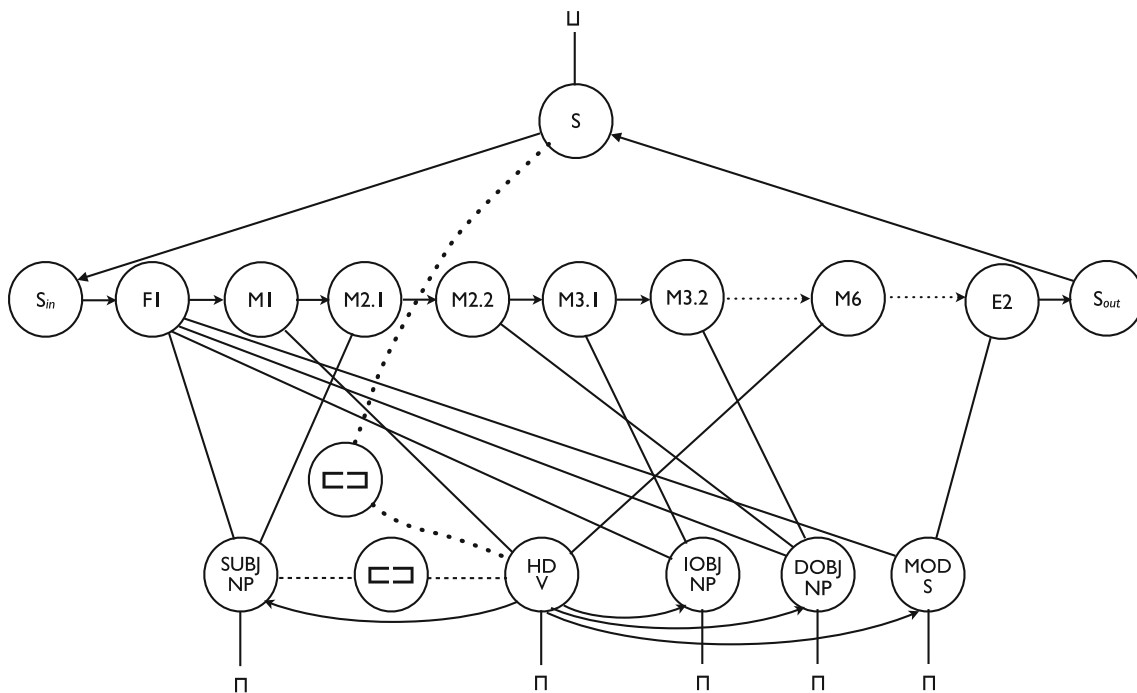
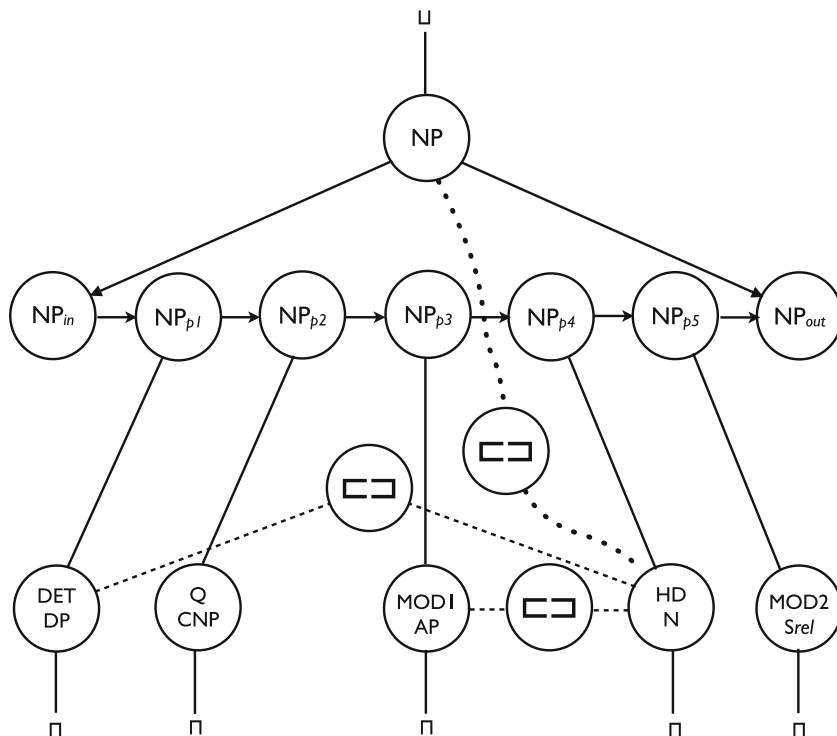


Fig. 5 Syntagma for decoding and encoding a simple Dutch main or subordinate clause. Some positions within the S-topology are not shown (dotted arrows). Pairs of square brackets (□□) denote unification circuits, to be introduced in the next subsection. Each single square bracket stands for a feature matrix that needs to unify with another matrix. The symbols \sqcap and \sqcup stand for feature matrices that need a hierarchically lower or higher unification partner. The curved dotted line serves to propagate morpho-syntactic features imposed by the bound (next-higher) syntagma to the Head verb (e.g., telling the verb to be finite, infinitival, or participial). Dashed lines indicate agreement within

the syntagma. The curved arrows at the bottom represent the effect of the verb’s subcategorization frame. For instance, a (di)transitive verb promotes the binding of Direct (and Indirect) Object(s) by sending some extra activation to the corresponding foot node(s); intransitive verbs send no extra activation. The early (“fronted”) positions of Direct and Indirect Object NPs—i.e., attached to topology position F1—are allowed, *inter alia*, if the NPs are bound by a relative or interrogative pronoun. Alternative linearization options inhibit one another (not shown). For instance, if the Subject NP has landed in slot F1, it inhibits the placement option of the Subject in slot M2.1

Fig. 6 NP-syntagma for Dutch. The unification nodes control agreement between Head noun, Determiner, pronominal Modifiers, and binding/bound syntagmas (these agreement relations are controlled by partly different feature matrices). DP=Determiner Phrase (often anchored by an article); Q=Quantifier; MOD1=Prenominal Modifier; MOD2=Postnominal Modifier



“Pluriclausal sentences and cross-clausal movement” we will see an example.

Furthermore, facilities are needed that force the RTN to wait in front of the position of an obligatory constituent, and allow it to skip optional constituents that are delayed.

Finally, the system should be able to recognize input constituents that have not been fully integrated into the emerged structure. Any constituent entering U-Space needs (1) a binding partner, and (2) a linear position. An activated place tag on the syntagma representing the constituent can signal fulfillment of the second need. The first need can be satisfied by a binary feature, called “*role tag*”, which marks whether or not the syntagma representing that constituent has found a (provisional) unification partner in a grammatical function/role “claimed” by the syntagma’s lexical Head. For instance, finite verbs may be said to claim a Subject, and transitive verbs a Direct Object; and all verbs can claim constituents in the grammatical roles of Modifier, Complementizer, and Subordinator. Role tags ensure that, at the end of the coding process, all active syntagmas have been claimed and are playing a permissible role in the hierarchical/functional structure. Role tags, like place tags, belong to the feature sets associated with root nodes of syntagmas, and are activated along with recruitment of the syntagma hosting them. Role tags differ from place tags only in that they become deactivated by the syntagma’s lexical Head rather than by the RTN. During the coding episode for a sentence, the recruited syntagmas keep transmitting activation and inhibition to other syntagmas as long as they are searching for optimal binding and placement alternatives. This continues until all role and place tags have been deactivated (or until they can no longer maintain a sufficient level of activation, or are stopped by an external force).

Features and Feature Unification

Many items of grammatical knowledge can be analyzed as matrices consisting of one or more features, with each feature made up of an (atomic) attribute and a set of currently applicable (atomic) values. Such matrices can be implemented in the form of *registers* (cf. Marcus 2009:163, quoted in the [Introduction](#) section). A register is a bundle of neural nodes, each representing the attribute and one value of a feature. If such a node is operational, the value is true; as long as the node does not fire, the value is false. If an attribute has more than one possible value, each of them is represented by a separate node within the register.

Let us assume that lemmas are represented by registers (feature matrices), and that activating a lemma means activating its register. Assume also that, in reaction to activation of a register, all and only those register nodes fire that denote attributes whose values are currently true of the lemma. Copying a register from its brain location to another location

is possible if each feature node in the original register is linked via a prewired connection to a node in the copy register. Actually making a copy means that, triggered via the prewired connection, all and only those nodes in the copy register start firing whose counterparts in the original register are currently true. (The firing pattern of the nodes of the original register does not change as a consequence of copying.)

Copying is needed, among other things, to transmit the feature matrices of activated lemmas from the Mental Lexicon (probably located in left temporal regions of the cortex) to the Unification Space (probably subserved by left inferior frontal cortex, in particular Broca’s area), and between registers within the Unification Space. Importantly, a lemma is not supposed to have an ID, i.e. a unique code that distinguishes the owner from all other lemmas and can be copied into an empty register elsewhere. *The identity of a lemma is its location in the network, i.e., its pattern of connections; this location does not change in the course of processing a sentence.* Among these connections are direct or indirect associations with other lexical items in the Mental Lexicon: *concepts* (relating lemmas to meaning) and *lexemes* (relating lemmas to phonological form). Consequently, copying a lemma does not involve copying an identifier/ID, but it means copying the register embodying its “lemma feature matrix.” (Likewise, concepts and conceptual structures will be represented in registers that have no ID either.)

When an empty register is filled with a copy of a feature matrix, the lemma that dispatched the matrix can only be traced back if there is additional circuitry that links the copy back to the original lemma and other information stored there (e.g. associated concepts and lemmas). Therefore, the copy register should transmit activation back to the original register via a connection in the opposite direction, at least for as long as the current sentence is being processed. Through such feedback connections, the system can retrace the path from any copy register (and even from a copy of a copy) to the original lemma.

Unification circuits (unifiers) function as a kind of filter between two registers—the unification partners. They are called into action when the registers of both partners are active and need to check whether their feature values are compatible. Unifiers decide not only whether or not the features of the unification partners match; they also relay the feature values resulting from successful unification back to the partners (and leave these values intact in case of unification failure).

Feature unification is defined as follows. A feature consists of an attribute (atomic) and a value. A value is a list of one or more (atomic) options (a disjunctive set). For instance, the English pronoun *you* has a morphological CASE value with three options (CASE = {nominative, dative, accusative}), whereas the CASE value of Subject NPs consists of one option only (CASE = nominative). Two exemplars of a feature with the same attribute can be *UNIFIED* (“are compatible”) if the intersection of their value sets is not empty. This intersection

is also the value of the attribute if unification succeeds (in the example: unification of *you* with a Subject NP yields “CASE = nominative” for both unification partners). If unification fails, the feature’s value sets remain unaltered. A feature *matrix* is a set of features with at most one exemplar of every attribute. Two or more feature matrices unify if all features shared by the matrices unify. This definition leaves open the possibility that the matrices to be unified share no feature at all (i.e. the attributes occurring in one matrix do not occur in the other matrix, and vice-versa), and unify trivially. In practice, however, candidate unification matrices nearly always share at least one attribute—often “Phrase/clause type” or “PoS”. Successful unification only affects the value sets of the unified features shared by the unified matrices: these sets now contain the intersections. However, matrices keep their original set of attributes.⁹

Microcircuitry for Feature Unification

Figure 7a sketches a neural network that accomplishes nonrecursive virtual feature unification as defined here, during both grammatical encoding and decoding. The circuit decides not only whether or not the features of the unification partners match; it also delivers the feature values resulting from successful unification (and leaves them intact in case of unification failure). These values are supposed to be adopted by—transmitted to—the unification partners (this is not shown in the figure).

Subject-Verb agreement in Dutch requires matching values of the Person and Number features, like in English. The bottom row of nodes represents values of the Subject unification partner—an NP headed by a noun or a pronoun; the row of nodes immediately below “verb” indicates the values of the other partner. The row of nodes in between are activated depending on whether the individual feature values of the partners are identical (i.e. match) or not. For example, the leftmost AND-gate node (within the “singular” subcircuit) will fire when both the verb and the subject (pronoun) include “singular” as a member in the value set of the Number attribute, thus yielding a value match. The right-hand neighbor of the AND-gate indicates value mismatch: It fires as long as there is not enough positive evidence for matching values in the unification partners, that is, until both partners have succeeded in activating the “Singular” node. As soon as this happens, the AND-gate gets active and sends sufficient inhibition to its neighbor to knock it out. Another consequence of activating the AND-gate is that one of the OR-gates at the top of the figure gets triggered as well. If such a scenario unfolds not only in one of the Number subcircuits but also in at least one of the three

Fig. 7 a Nodes and connections within a feature unification circuit for Subject-Verb agreement. *Circles* inscribed with the \wedge -symbol are AND-gates; circles with the \vee -symbol are OR-gates. The five identically shaped subcircuits determine presence or absence of matching values of the Person and Number attributes. When the module is called into action, the nodes drawn with *bolded lines* are immediately activated, indicating that no unification has taken place. **b** Continuation of the scenario started in **a**. The feature values of the words *geld* and *telt* have been added. *Geld* has singular Number and 3rd Person (like *money*); *telt* is singular but its Person can be 2nd or 3rd. **c** The AND-gates of singular Number and 3rd Person have fired, suppressing their right-hand neighbors and activating both OR-gates in the top: The unification succeeds. **d** In three of the five feature values there is insufficient evidence for matching values; therefore, the “mismatch” nodes in their subcircuits are not suppressed, and inhibit the feature value nodes of both unification partners. This affects the “2nd Person” node of *telt*, causing this verb to retain only its “3rd Person” value. The remaining active/applicable values are transmitted to the registers representing the unification partners (not shown). **e** Unification during grammatical encoding. Continuation of the scenario started in **a**. The meaning representation underlying the Subject has been lexicalized as the lemma *geld* with singular Number. For the verb lemma, all Number and Person values are still open. The scenario ends as in **d**: The AND-gates of singular Number and 3rd Person will fire, suppressing their right-hand neighbors and activating both OR-gates in the top: The unification succeeds. “Mismatch” nodes inhibit the feature value nodes of “plural Number”, “1st Person” and “2nd Person”. The verb form (lexeme) *telt* expresses the resulting feature values

“Person” subcircuits, the topmost AND-gate fires, meaning that the partners unify successfully.¹⁰

Figure 7b–e detail the subject-verb unification process that takes place during decoding and encoding of example sentence (2). Both instantiations of the process start with the activation pattern in Fig. 7a, and end with the one in Fig. 7d; some intermediate steps, shown in Fig. 7b–c (decoding) and e (encoding), differ slightly between coding modalities.

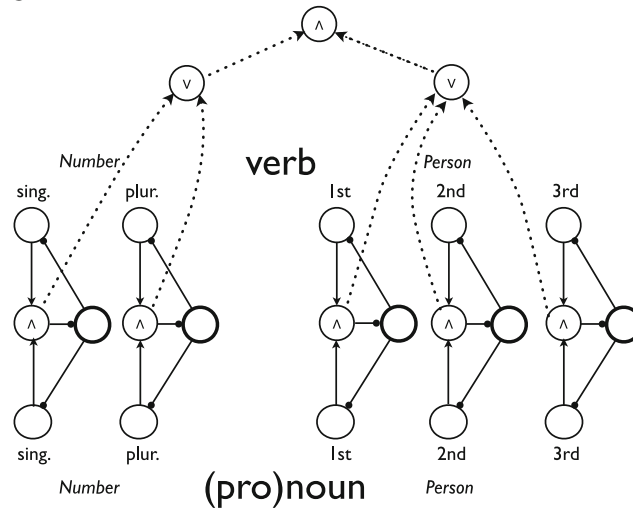
- (2) Geld telt
‘Money counts’

The definition of feature unification allows for the possibility that one partner possesses features (attributes and values) that have no counterpart in the other partner. For instance, NPs have a morphological Case feature (nominative, dative, etc.) but verbs do not. The fact that unification circuits are components of prewired (preconfigured) neural connections between registers that represent the unification partners, enables fine-tuning the selection of features participating in a unification. Connections standing for attributes that do not participate in the unification at a given position in the network, are simply not attached to the unification circuit. For example, although the Case feature appears in the unifier that binds the Subject NP of a verb (the foot node that dictates nominative Case) to a specific NP (the root

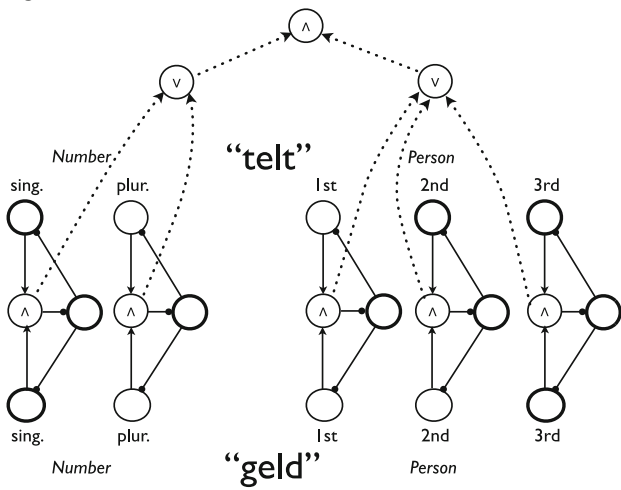
⁹ This unification operation differs from other unification operations discussed in the literature in several respects. It is *nonrecursive* because values of features are atomic and cannot be feature matrices themselves; and it is *virtual* because the unified matrices remain distinct objects and do not “merge” (De Smedt 1990). In the model, successful unification usually entails opening of a gating circuit and the start of activation (cf. resonance) flowing between the nodes that “own” the matrices.

¹⁰ I will not pursue details concerning the temporal and other dynamic requirements to be met by the flow of activation and inhibition in unification circuits.

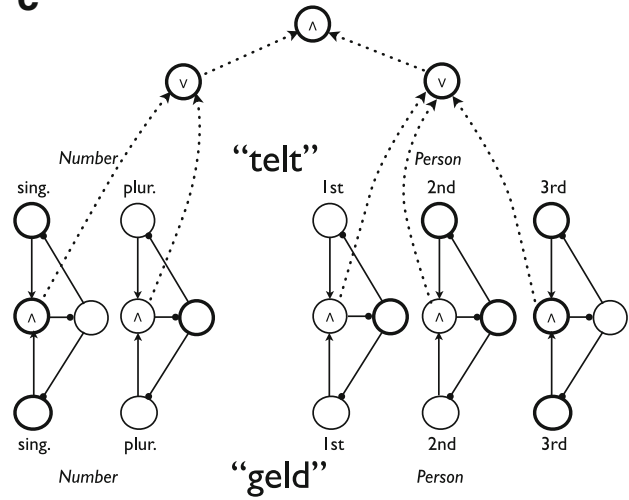
a



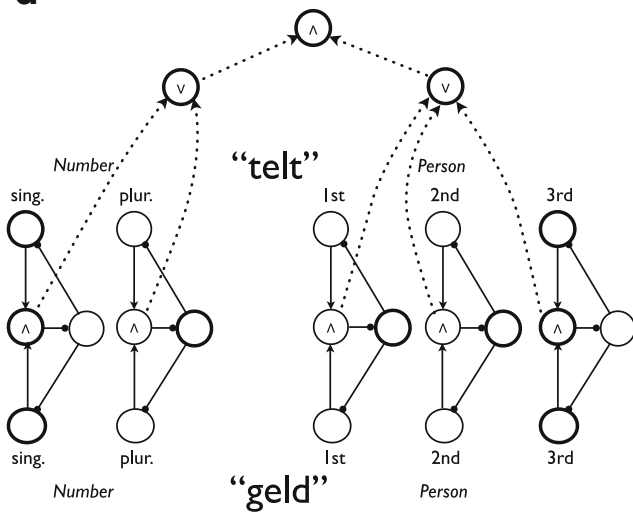
b



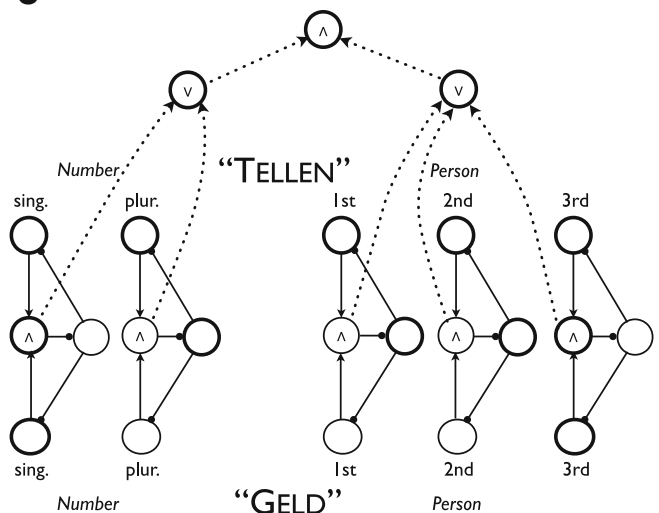
c



d



e



node that thus inherits nominative Case of the Subject NP), it does not appear in the unifier for subject-verb agreement.

The unification microcircuits are located between the to-be-unified matrices (or rather the registers representing the matrices). For instance, Subject-verb agreement takes place in a unifier that connects the matrices of a Subject NP and a finite verb (cf. Fig. 5). The wiring of the unifier determines which features are taken into account—in English and Dutch: Number and Person. Any other features—e.g., nominative case of the Subject NP—bypass the unifier circuit and are left out of consideration.

As a graphical notation for unification circuits I use pairs of rectangular brackets facing one another ($\sqsubset\sqsupset$), where each bracket denotes the matrix of a unification partner (as announced in the caption of Fig. 5).

Winner-Take-all Competition

The machinery introduced so far allows a more accurate description of the neural connections subserving winner-take-all competition (see the SINUS model in Vosse and Kempen 2009a, for a computational implementation).

Figure 8 illustrates the crucial connections needed to handle the competition between three NP-syntagmas that attempt to bind three major NP-constituents of a finite clause (an S-syntagma) during a decoding episode. Any NP can only fulfill one grammatical function, and any function can only be fulfilled by one NP. The system presupposes that the NP-syntagmas are recruited by the NP router sequentially. Hence, if the unifications succeed, NP1 will first bind the Subject, then NP2 the Indirect Object, and finally NP3 the Direct Object—in agreement with the schedule followed by the RTN. The web in the middle of the figure shows that the NP-syntagmas and the foot nodes of the S-syntagma are fully connected, yielding ample opportunities for “wrong” bindings to emerge. However, the inhibitory links render this very unlikely. For instance, once the binding of NP1-to-Subject has been established,¹¹ the corresponding link starts emitting inhibition to four other links. Two of these will also be activated while the NP2-to-IOBJ binding emerges: one attempts to bind the Subject to NP2, the other one the Indirect Object to NP1. However, this competition does not jeopardize the emerging IOBJ-to-NP2 binding because the two competitors are inhibited by the strong Subject-to-NP1 binding. Obviously, this course of events presupposes adequate parameter settings determining the amounts of activation and inhibition transmitted via the

¹¹ Just as in the SINUS model, I assume that the first position of the S-RTN can be occupied by at most one constituent, but various types of constituents are eligible, including constituents fulfilling roles other than that of Subject—as depicted in Fig. 5. Psycholinguistic experiments suggest, however, that the coding system has a preference for early placement of the Subject. In the proposed model (and in SINUS) this effect is reached by initializing the Subject foot node of recruited S-syntagmas at a higher activation level than that of other foot nodes.

various links.¹² Another factor conducive to the emergence of the desired bindings is the presence of unifiers in the links between root nodes and foot nodes. Furthermore, the figure assumes that unification failure eliminates the resonance effect (or reduces it drastically), so that the amount of inhibition transmitted to competitors also drops considerably.

The wiring depicted in Fig. 8 raises a crucial question: Does the portrayed activation and inhibition scheme scale up to larger numbers of NP- and S-syntagmas? A more realistic coding network needs more than three NP-syntagmas and more than one S-syntagma, and the danger of a combinatorial explosion of the number of inhibitory links between all potential bindings seems immanent. For instance, if the network would include ten NP-syntagmas and three S-syntagmas, the number of activation links (potential bindings) of these NPs with three foot nodes in each S-syntagma would amount to $10 \times 3 \times 3 = 90$; the nine binding options of each NP root node would require $9 \times 8 / 2 = 36$ inhibitory links; and the ten binding options of each foot node of the S-syntagmas would even require $10 \times 9 / 2 = 45$ inhibitory links. However, although the model indeed assumes full excitatory connectivity between the root nodes of all syntagmas with all qualified foot nodes of any other syntagma, in practice there is no need for an exhaustive set of inhibitory connections between any pair of excitatory links that in theory might enter into competition. The reason has to do with the fact that recruitment of syntagma exemplars (under router control) and the RTNs both operate sequentially.

To see this, consider Fig. 8 again, and assume that the three NP-syntagma exemplars are the first three of a supply list of ten exemplars that the NP-router recruits *in a fixed order*. This entails a statistical tendency for NP exemplars nearby in the supply list to bind foot nodes nearby in RTN trajectories. In the example, if the first NPs would have bound foot nodes of an earlier S-syntagma, the depicted S-syntagma would probably have been bound by—and aligned with—three consecutive members further down the supply list of NP-syntagmas. Hence, inhibitory links are only needed from each bindable foot node to a few (maybe three or four) consecutive members

¹² The parameter values for SINUS were estimated during a training period in which the model received a sample of target language sentences that are known to elicit human parsing phenomena (e.g., a typical “garden-path” sentence, and a nearly identical counterpart that does not lead the reader up the garden-path; a set of sentences with center-embedded versus right-branching relative clauses; sentences with or without a word-class ambiguous lexical item at a certain syntactic position, etc.; for details, see Vosse and Kempen 2009a). A simulated-annealing procedure was used to find, in the total parameter space, a subspace where the model exhibited the psycholinguistically expected performance (e.g. cognitive processing load measured in terms of processing cycles in the model, or in terms of the proportion of unsuccessful parsing trials). After the training episode was completed and the parameters fixed, the model was tested on syntactic structures it had not seen before. It remains to be seen whether this parameter estimation procedure can be adapted to the encoding modality, and/or whether one set of parameter values can satisfactorily simulate decoding as well as encoding performance.

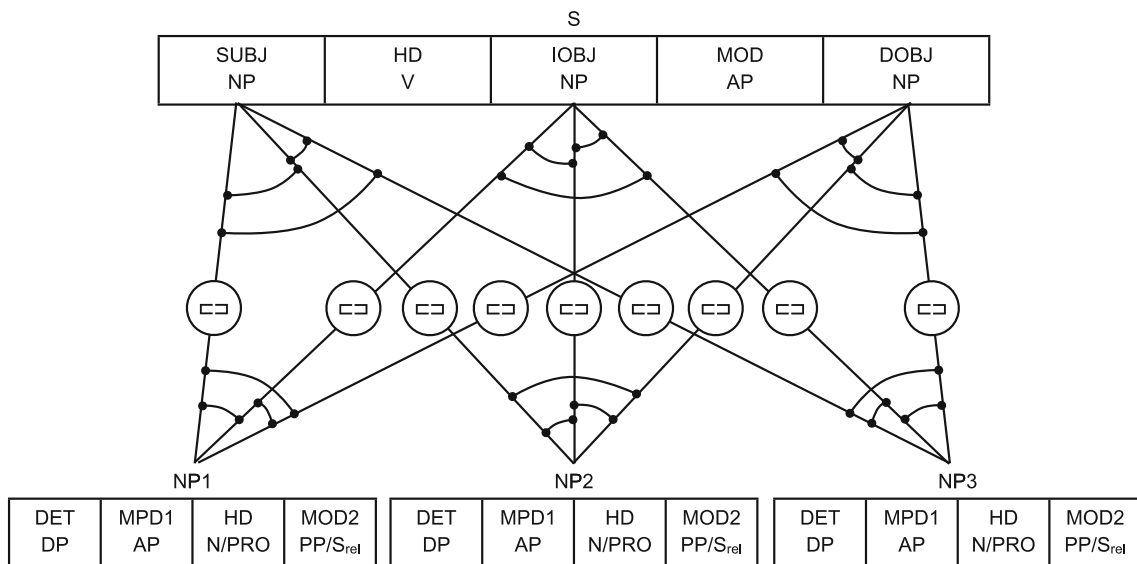


Fig. 8 Winner-take-all competition between three NPs attempting to bind the Subject, the Direct Object, and the Indirect Object of a clause (e.g., *Peter gave his girlfriend the latest smartphone*). The rectangle at the top depicts some major constituents of the clause in their standard linear order; the three rectangles at the bottom show important

constituents of NPs. The circles are unification circuits (e.g., checking nominative, dative, or accusative case of prospective unification partners; and, in case of success, communicating the resulting case value to the partners). Straight lines: bidirectional excitatory connections representing possible bindings; curved lines: bidirectional inhibitory connections

of a supply list of same-type syntagma exemplars (e.g., between NP1–NP2–NP3, between NP2–NP3–NP4, and between NP3–NP4–NP5 but not between more distant members NP1–NP5–NP9). Similarly, each NP-syntagma exemplar has the potential to bind not only the three foot nodes of the S-syntagma depicted in the figure, but also the corresponding foot nodes of (at least) two other S-syntagmas from the stock. This requires nine activation links for each NP exemplar, but not $9 \times 8/2 = 36$ inhibition links: Nine inhibitory connections probably suffice. For example, NP1 would need the three inhibition links shown in the bottom-left corner of the web in Fig. 8, and two similar inhibitory triplets between activation links running from NP1 to foot nodes of the two other S-syntagmas. In conclusion, the threat of a combinatorial explosion of the number of inhibitory links when the network is scaled up, is considerably reduced due to the expected temporal proximity of bindable foot nodes on the one hand, and the temporal proximity of binding root nodes on the other.¹³

In the scenario depicted in Fig. 8, the three unifications/bindings were fortunate in that their first choice of a binding partner turned out to be the ultimately correct one. In less fortunate circumstances, however, subsequent events may call for a revision. Nonstandard linear orders of grammatical functions are a case in point. In relative clauses, for instance, it is the

constituent consisting of—or containing—the relative pronoun, that opens the clause. However, this constituent may not play the role of grammatical Subject (as in *The woman [that the boy insulted] was not angry*). The RTN within the relative clause will initially accept the relative pronoun as Subject—but not for long: The next position of the RTN cannot host an NP such as *the boy*, thus preventing the RTN to proceed. However, the NP-syntagma bound by *the boy* keeps sending activation to all its potential unification partners, including the Subject foot node. Meanwhile, the NP-syntagma bound by the relative pronoun tests its odds with other foot nodes of the current syntagma. One of them fulfills the role of Direct Object, which is “fronted” if bound by a relative pronoun (in English as well as in Dutch; cf. Fig. 5). The link representing this binding inhibits the competing binding between NP *that* and the Subject foot node, thus supporting NP *the boy* to gain Subjecthood. The end of this scenario is an optimal solution, allowing the RTN to transit to subsequent positions, with *that* as Direct Object (ultimately licensed and supported by the transitive verb *insulted*) and *the boy* as Subject. (I refer to Vosse and Kempen 2008, 2009a for computer implementations of Dutch and German parsing-with-revision scenarios similar to the English one sketched here.) Notice that no exchange of binding partners would have taken place if word order had been slightly different. In *The woman [that insulted the boy] ...*, the relative pronoun can keep its status of Subject uncontestedly all the way through the clause. Indeed, psycholinguistic experiments observe that, in many circumstances, “Subject relative clauses” are somewhat easier to process than “Object relative clauses” (e.g., Staub 2010; Lewis and Vasishth 2005).

¹³ Needless to say, these calculations are extremely tentative, not only because, as already indicated, the basic numbers of syntagmas needed are rough estimates, but also because additional factors may be at work, in particular lexical and syntactic ambiguity (increasing the demand) and conceptual factors (decreasing it).

What happens when a winner-take-all competition remains unresolved, that is, when none of the contestants manages to defeat its adversaries? Consider the situation of an RTN that, during the traversal of a topology, enters a slot that is to be filled by an obligatory constituent. An example could be the slot where the Subject foot node of a finite clause has to land. If more than one NP is trying to bind that foot node but no winner has shown up yet, the RTN seems to have no option but to await the emergence of a winner. However, let us hypothesize that the RTN, in order to lift such a blockade, has a means to *coerce* a lingering competition to resolve. One such means involves adding random amounts of activation to the mutually inhibitory links through which the competitors contend with one another (“disinhibition”). These amounts should be small initially but grow steadily. This increasing noise guarantees that, after some interval, the difference between the activation levels of the competitors is big enough to yield a winner. Clearly, a winner thus selected may turn out to lead the system into just another blockade (a “garden path”). But the advantage is that the RTN can continue its traversal and take new input into processing—input with potential repercussions on the competitions underlying recent bindings (as illustrated in footnote 8).

Pluriclausal Sentences and Cross-Clausal Movement

The treatment of sentences containing more than one clause requires some extensions of the model, in particular with a view to the treatment of cross-clausal “movement” of syntactic constituents: the phenomenon that constituents sometimes seem to escape from the clause they belong to and to occupy a position in a clause higher up in the hierarchy of clauses. In example (3a), the interrogative constituent *which disease* occupies a position in the main clause headed by *did*, not in the subordinate clause headed by *contract*, where it plays the role of Direct Object. Cross-clausal movement is restricted to declarative Complement clauses; in English, they optionally start with the subordinating conjunction *that*. No constituent can escape from *interrogative* Complement clauses (as evidenced by the ill-formedness of (3b)), or from adverbial clauses (cf. (3c); adverbial clauses begin with a subordinating conjunction other than *that*—here with *because*—, and function as Modifier within another clause).

- (3) a. Which disease did you say (that) John contracted?
 b. *Which disease did you ask who contracted?
 c. *Which disease did John go to the hospital because he contracted?

Sentence (3a) contains three verbs, each binding an S-syntagma, with *did* at the top and *contracted* at the bottom of

the hierarchy. The verbs *did* and *say* each take a Complement in the form of another clause. The connections are realized by direct unification links between a Complement (CMP) foot and the root of an S-syntagma. I now describe—first in a decoding, then in an encoding episode—how, in sentence (3a), the interrogative NP gets its high/fronted position (see Fig. 9).

Underlying the figure is a basic principle that we have already met several times: *full connectivity*. All permissible root-to-foot bindings between the syntagma exemplars in the system are present as prewired connections. For the example, this implies that the NP *which disease* sends activation to all NP foot nodes of all S-syntagmas in the system, but that initially only one of them resonates—the top S-syntagma which, as binder of the Apex, become the main clause. In this syntagma, the NP is able to bind a foot node that allows its binding partner to land in F1 (as interrogative NPs nearly always land in F1 slots). The most likely winning foot node is the Subject. After *did* has become the lexical anchor, NP *you* also starts competing for the Subject role. As it soon becomes clear that *did* subcategorizes for a Complement clause rather than for a Direct Object, NP *you* wins the competition for Subjecthood, thus leaving the Wh-NP in limbo, unclaimed. However, new input keeps coming in and the RTN cannot lag far behind. This triggers a *coerced* binding (as described at the end of the previous subsection) of the Wh-NP into the role of Direct Object (which presumably is more frequent than Indirect Object). The RTN now proceeds its course through the topmost topology and applies readout to *Which disease did you*. (Readout during decoding serves to check parsed word order against actual input word order; more on this in the next section).

Crucial ingredients in the treatment of grammatical movement are the place and role tags introduced above (see end of section “*Syntagmas*”). Under certain circumstances, a syntagma may receive its role and its place tag from two different syntagmas. This is what happens in example (3a). The finite transitive verb *contracted* “claims” the Wh-NP as its Direct Object, although this constituent has already received a place tag from the RTN in the upper syntagma. As neither *did* nor *say* claim a Direct Object, the lower verb *contracted* can defeat the upper S-syntagma, causing the Wh-NP to be attached low—“low” in terms of root-to-foot binding as well as topology placement. (The relocation is covert because the Wh-NP has been read out before.) Note that the new landing site of NP *which disease* does not yield a conflict with any other constituent that might need to land there: Except for a Wh-constituent, no other constituents of declarative Complement clauses select F1 as destination. In consequence, NP *which disease* ends up binding a foot node of the lower S-syntagma while overtly it seems to occupy a topology slot in the upper S-syntagma.

The proposed mechanism for “Wh-extraction” also accounts for the ungrammaticality of variants (3b) and (3c). In (3b), the middle S-syntagma is anchored by the verb *ask*, which takes an interrogative Complement. However, the F1

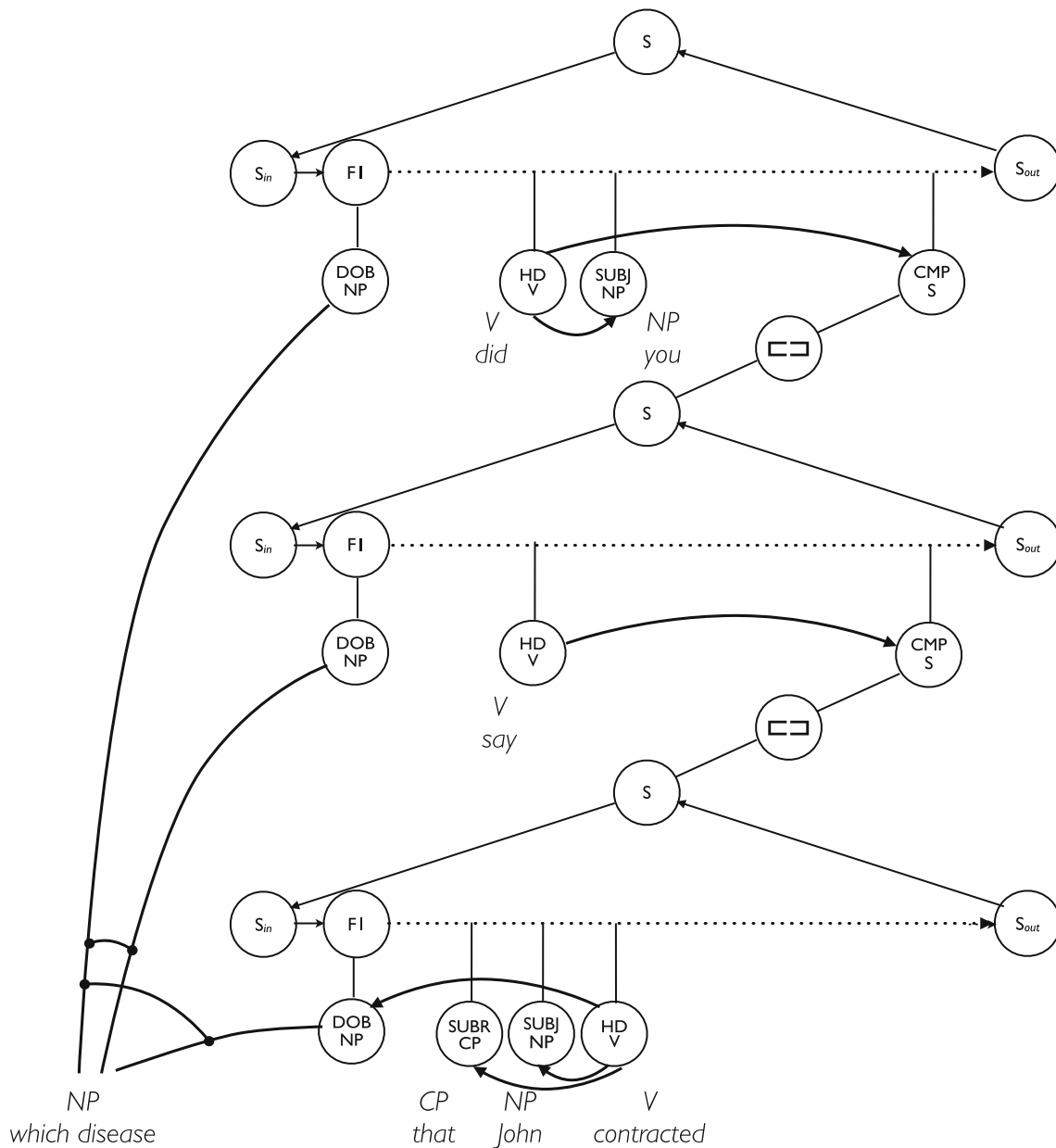


Fig. 9 Hierarchy of S-syntagmas for example (3a). The *triangular shape* at the top, with an S-root node and nodes S_{in} and S_{out} as vertices, represents an S-syntagma. The topology is indicated by a *dotted arrow*; only the first slot, F1, is shown. Two similar triangles are depicted in the middle and at the bottom. Below each topology is a selection of foot nodes that can be bound by a lexical item or another syntagma. The Head verb of the top S-syntagma (*did*) claims a Subject and a Complement, as indicated by the *curved arrows*. All S-syntagmas are assumed to have the same standard shape, including the same set of foot nodes.

Each of these foot nodes is connected, via prewired links, to all (examples of) syntagma types that can bind them. Three of these mutually inhibitory links run between the bottom-left NP-syntagma (*which disease*) to three Direct Object NP foot nodes (*curved continuous line* ascending from this Wh-NP; the unification circuits have been left out). Similar links—left out as well—run from this (and other) NP-syntagmas to Subject and Indirect Object foot nodes of all S-syntagmas: full connectivity. However, only foot nodes of recruited S-syntagma will effectively become bound

slot of interrogative Complements is always occupied because F1 is the typical landing site of Wh-constituents (*who, what, whether, why, etc.*). Therefore, NP *who* will compete for a binding in the lower S-syntagma and, if successful, occupy the F1 slot there, with both its role and its place tag deactivated. Competitor NP *which disease* has to stay behind at the F1 of

the upper S-syntagma, with a deactivated place tag but an unclaimed and thus still active role tag. Example (3c) gives rise to a similar scenario. F1 slots of adverbial clauses are the obligatory landing site of the subordinating conjunction exercising the role of Subordinator (here: *because*). Hence, NP *which disease* cannot be covertly relocated.

As is well-known, the level of acceptability/grammaticality of sentences embodying Wh-extraction depends strongly on the choice of Complement taking verb (compare *Which disease did you say that John contracted?* with *Which disease did you mumble that John contracted?*). Furthermore, Wh-extraction is not licensed by Complement taking *nouns*. The sentence *Which disease did you claim that John contracted?*, with *claim* as verb, is much more acceptable than *Which disease did you make the claim that John contracted?*, with *claim* as noun. The model can explain these effects by postulating that the F1 slot in the S-syntagma recruited to build the finite Complement clause is not always fully activated. More specifically, assume that the syntagma bound by the Complement taking verb or noun is able to influence the level of *activation* of the F1 slot in the S-syntagma that represents the Complement clause. The influence can be exercised via the unification link between the syntagmas involved. Strong activation of the neuronal assembly representing the F1 slot facilitates the emergence of a “low” binding of the Wh-constituent (in terms of Fig. 9: the binding within the lower S-syntagma). If, upon recruitment, this neural assembly receives little activation, or is strongly inhibited, the low binding is difficult or impossible. This has the same effect as competition for the F1 slot with another Wh-constituent, as in (3b) and (3c).

Behavioral evidence supports this account. A verb that yields high-grammatical Wh-extraction sentences, activates the entire topology of the Complement clause, including the F1 slot. In German, finite Complement clauses are sometimes realized in the form of main clauses, i.e. with the finite verb in “Verb-second” position. This is illustrated by example (4a), from Featherston (2004:187); in (4b), the Complement clause embodies the “Verb-final” word order which is standard in subordinate clauses. Complement clauses such as in (4a) need a full topology to provide a landing site for the Subject NP (*er* ‘he’). In a grammaticality judgment experiment, Featherston (*o.c.*) compared the grammatical quality of sentences embodying Wh-extraction with sentences containing a main-clause Complement (i.e., (4a)), using the same set of Complement taking verbs in both structures. He observed a high correlation between the two sets of judgments: If a verb yields high-quality Wh-extraction structures, it also yields high-quality main clause Complement structures, and analogously for the low-quality structures. These data provide support for the crucial assumption about the high activation level (free availability) of the F1 slot of declarative Complements of verbs that facilitate Wh-extraction.

- (4) a. Max hofft, er hat die Geschworenen überzeugt
 Max hopes he has the jury convinced
 ‘Max hopes he has convinced the jury.’
 b. Max hofft, dass er die Geschworenen überzeugt hat
 Max hopes that he the jury convinced has
 ‘Max hopes that he has convinced the jury.’

Gibson and Warren (2004) published a reading time study that enabled them to compare the processing load required by sentences with vs. without Wh-extraction. The sentences were comparable in respect of syntactic complexity and semantic content. The authors interpreted the pattern of reading times for the two sentence types as evidence for “intermediate syntactic structure”. The data suggested that, located immediately before the syntactic position of the Complementizer *that*, there was some sort of steppingstone that facilitated the syntactic and semantic integration of the extracted Wh-constituent with the verb claiming it as Direct Object (in (3a): the integration of *which disease* and *contracted*). They identified this position as—in the terminology of mainstream generative grammar—the specifier of the CP of the embedded clause. This is the position that, in the present model, corresponds to the F1 slot of the Complement clause. Apparently, the availability of this position enables the “low” binding of the Wh-constituent and, via this binding, its semantic/conceptual integration with the Complement’s Head verb.

Next, I take up the grammatical *encoding* of sentences with Wh-extraction such as (3a), focusing again on the placement and the binding of the Wh-constituent. I do this while making minimal assumptions about the conceptual structure specifying the speaker’s intention. Also, I will not describe in detail how the U-Space assembles the syntagma hierarchy, except for announcing that a crucial role is played here by *referential indices* associated with the input lemmas (see the next section for the neural correlates of these indices). For instance, NP *you* will preferentially bind the Subject foot node of *did* or *say* because this NP has an index that matches the index of the Subjects of *did* and *say*. However, because *did* instructs its Complement (via the unification link) to be infinitival, NP *you* cannot bind the Subject foot node of *say*, due to inhibition of this node. This leaves the Subject foot node of the upper syntagma as the only suitable binding partner for NP *you*. As part of the root-to-foot binding process, the referential indices are compared. If they match, the binding circuit receives extra activation (positive auto-associative feedback), enabling it to inhibit and defeat other binding options explored simultaneously via a prewired link. E.g., although the NP-syntagma bound by *you* has a prewired connection with the Subject foot node of the S-syntagma bound by *contracted*, this binding will be rejected immediately due to the mismatching indices, and succumb to inhibition from the binding that does have matching indices.

How does the interrogative constituent *which disease* receive its prominent place at sentence onset, at the top of the S-syntagma hierarchy, in spite of referential indices telling this Wh-NP that it belongs to the lower S-syntagma, and should play the role of Direct Object there? The answer presupposes that the input to the U-Space always specifies the value of the “mood” feature of the various clauses. Possible values are

declarative, interrogative (Y/N- or Wh-question), or imperative. They are represented by dedicated neural assemblies within S-syntagmas—here “Wh” and “declarative” for the main and the embedded clause, respectively. This activation spreads to other assemblies influencing the shape of the clauses. In particular, the mood value Wh-question leads the RTN to demand a Wh-constituent as filler of the F1 slot, and to move to subsequent slots only after one has appeared. If several constituents present themselves as candidate fillers, the RTN is triggered to coerce a binding by adding random amounts of activation to the candidates, until one has managed to suppress the others. For sentence (3a), this implies that readout of the sentence will commence only after Wh-NP *which disease* has occupied the F1 slot of the main clause. The remainder of the encoding process yielding (3a) resembles the decoding process for this sentence, and includes the covert relocation of the Wh-NP binding from the upper to the lower S-syntagma. Differences between the encoding and the decoding scenario are probably restricted to the order in which the bindings take place. In decoding mode, this order depends on the input order of lexical items; in encoding mode, the order is determined rather by conceptual and lexical factors.

A Neurocognitive Architecture for Grammatical Coding

The Unification Space is not the only neurocognitive module involved in grammatical coding. The present section explores how grammatical coding in the U-Space interacts with other language production and comprehension processes, and what the proposed model entails with respect to the neurobiological infrastructure.

Referential Processing and the Hippocampus

Referents are entities (objects, animals, humans, or other “protagonists”), states (properties), state changes, or actions perceived and recognized in the real or imagined world. Combinations of protagonists with the states, state changes or actions in which they are involved, are the nonlinguistic “flesh and blood” of communicative intentions. The set of referents that has been addressed since the onset of the current dialogue is called the “referential domain” of that dialogue. The maximum number of referents that inhabits the referential domain at any given point in time, is restricted—not larger than, say, a hundred. Referents are internally (cognitively) represented in the form of *mental models* (Johnson-Laird 1983, 2006), or *schemas* (Arbib 1989), of the described situations and events (see Glenberg et al. (1987) for a compelling behavioral demonstration, and Barrès and Lee (2013) for a computational implementation based on schema theory). Mental models are *iconic* internal representations of aspects of the external reality, that is, of the referents mentioned in a sentence. Here, the term

“iconic” indicates that each part of the model, and each relation between parts, corresponds to a part and a relation, respectively, in the represented reality. Mental models are abstract in that they represent reality in a viewpoint-independent manner (allocentric); in this respect, they differ from images and pictures, which are viewpoint-dependent (egocentric). Mental imagery is egocentric *and* iconic (see Johnson-Laird 2006, for definitions and many examples).

The composite mental model created for an entire input sentence or clause is composed of mental models triggered by the individual words. During comprehension episodes, mental models associated in long-term memory with the individual words are retrieved and combined into a composite mental model for the described scene. (Not all aspects of the meaning of a sentence can be captured by a mental model—negation is an example. Such aspects can be represented by annotations attached to the composite mental model.) Johnson-Laird (2006) emphasizes the importance of *spatial* representations, in particular when the to-be-processed text embodies reasoning tasks.

An important role in the service of visuo-spatial processing is played by the *hippocampal complex* (HC) which provides for *rapid online storage* of visuo-spatial experiences. This functionality rests on the high level of synaptic plasticity in this region—much higher in the HC than in the neocortex (cf. the “complementary learning systems” framework proposed by McClelland et al. (1995); for new computational implementations of this framework, see Kumaran and McClelland (2012), and Howard et al. 2011). A recent neurocomputational model of hippocampal functioning proposes detailed neuronal mechanisms not only for online memory storage of perceived visuo-spatial scenes but also for imagining novel scenes (Byrne et al. (2007; see also Bird and Burgess 2008). The latter task involves the online creation of novel associations between stored representations of objects on the one hand, and spatial locations on the other.

This suggests that the hippocampal complex can also subserve the online creation of composite mental models representing (major aspects of) the meaning of novel sentences. Stated differently, the combinatorial aspects of referential processing during sentence comprehension can do without *prewired* connections between all possible combinations of mental models associated with nouns, verbs, and members of other word classes. Apparently, while the words of a sentence come in, the hippocampal complex can access, in real-time, exemplars of the mental models associated with to these words in long-term memory (residing in temporal and other cortical regions), and establish new associations—“conjunctive bindings”—between them (with the precise form of these associations depending on the syntactic structure of the input sentence). The resulting conjunctive bindings allow the construction of integrated visuo-spatial mental models, which embody the *referential structure* of the input sentence. Therefore, the ban on the online formation of novel neuronal connections

(discussed in the [Introduction](#)) does not apply to the *referential* structure of sentences.

Referential indices are temporary identifiers (IDs) of mental models (i.e., of the internal representations of referents). A given referential domain may include more than one token of the same mental model. One way of telling them apart is to associate them with different spatial positions. This is how the Hippocampal Complex (HC) solves the problem, utilizing its rapid learning capacity. In real-time, it creates temporary associations between perceived objects and spatial locations (see, e.g., the review by Bird and Burgess 2008, of spatial processing in the HC). Theories of episodic memory often postulate a generalized version of this mechanism, according to which the HC can rapidly establish associations between objects and *any* type of *context* properties (see Diana et al. 2007, and the references cited there). These theoretical frameworks justify the idea that the HC forms online associations between mental models (i.e., internal representations of referents) on the one hand, and *contextual coordinates* on the other, thus providing the mental models with referential IDs. I will assume henceforth that each mental model activated during an encoding or decoding episode is associated with a unique set of contextual coordinates, and that these coordinates are temporarily attached to the lemma(s) that, in the target language, designate(s) that mental model. In line with (psycho)linguistic conventions, I call them *referential indices*.

Upon first mention during a dialogue, the individual referents—or rather the mental models representing them—are thus bound to a unique index. The bindings are retained during the current encoding and/or decoding episode, so that they are accessible from other processing modules, e.g. to check coreferentiality of two NPs.¹⁴ Referential indices also provide the key to a solution of the *symbol-grounding problem*: They enable the language user to link linguistic symbols—lexical concepts and the corresponding annotated lemmas (see below)—to their referents in the external or imagined world (Harnad 1990; Barsalou 1999). When the episode ends, the bindings are cleared to make room for a new referential domain. In this paper, symbols from the set {i1, i2, i3, ...} will serve as notational substitutes for the contextual coordinates provided by the Hippocampal Complex.

Annotated lemmas are registers that specify not only the lemma's morpho-syntactic features (long-term-stored in the Mental Lexicon), but also lemma properties related to—and valid only during—the current dialogue (encoding/decoding

episode). These include the lemma's own referential index, and indices of governed lemmas. A simple example (cf. sentence (3a)): Suppose the lemmas underlying the verbs *say* and *contracted* are assigned the referential indices i1 and i2, respectively; and the NPs *which disease*, *you*, and *John* are indexed by i3, i4, and i5. Then, if we want the U-Space to bind *contracted* as Head of the Complement clause governed by *say*, we only need to associate the referential index i2 with the Complement foot node of *say*. After that, any attempt to bind the Complement of *say* fails if the would-be binder (e.g., *by did*) does not carry the matching index i2. In the same manner we can force the U-Space to only accept i3 as Direct Object of *contracted*, etc. Therefore, despite full connectivity (e.g., all NP-syntagma exemplars are connected with the Subject and the Direct Object foot nodes of all S-syntagma exemplars), the system can still create the intended bindings by selectively activating bindings between partners with matching referential indices.

Functional and Conceptual Dependency Graphs

The grammatical decoding process within the Unification Space delivers structures that I will refer to as “*functional dependency graphs*”: They specify, as part of the resulting lemma annotations, the functional-grammatical relations between the lemmas involved. The next step in the comprehension process consists of the conversion of functional dependency graphs to “*conceptual dependency graphs*”. This requires mapping lemmas and the functional dependencies they govern, onto the concepts and conceptual dependencies stored in entries of the Mental Lexicon. For instance, the Subject dependency governed by the verb *give* expresses the actor participating in a GIVE act; but in case of the verb *receive*, the Subject indicates the recipient. Following standard (psycho)linguistic practice, I assume that the translation from syntactic to conceptual (also called thematic) structures is an important step in the semantic/pragmatic interpretation of input sentences. Importantly, the functional-to-conceptual conversion includes copying of referential indices associated with the lemmas onto the corresponding concepts.

Conceptual dependency graphs also play a crucial role during grammatical *encoding*. They represent an early step in the process leading from the speaker's communicative intention to a spoken utterance. As part of this process, conceptual dependency graphs are mapped onto functional dependency graphs, which then enter the Unification Space as a sequence of referentially and morpho-syntactically annotated lemmas.

U-Space Input and Output: The Lemma-Syntagma Interface (LSI) and I/O Buffers

U-Space activity is driven by registers representing annotated lemmas. They are entered there via an interface component, called Lemma-Syntagma Interface (LSI), that connects

¹⁴ Language pathology also points to the role played by the HC in the referential aspects of grammatical coding. Patients with hippocampal amnesia appear to use indefinite Noun Phrases, e.g. *a windmill*, to refer to just-mentioned objects that would normally be described in terms of a definite NP (*the windmill*; example from Duff et al. 2011; see also Duff and Brown-Schmidt 2012). MacKay et al. (2013a, b) describe in detail the language deficits of neurological patient H.M. who was amnesic after bilateral surgical removal of large parts of the hippocampal complex.

the Mental Lexicon to the Unification Space in both directions. The LSI is a two-dimensional (word position * syntagma type) array whose cells are registers into which lemma feature matrices can be copied. An LSI with fifteen positions and seven syntagma types can handle sentences of up to fifteen words long (even longer ones if early, decaying syntagmas can be reused); and each of the seven cells at a given position provides a binding opportunity with one syntagma type (S, NP, DP, AP, PP, CP, or CNP). All cells in the array include a unification circuit that checks compatibility of two matrices: the feature matrix associated with an annotated lemma adduced from the Mental Lexicon, and the feature matrix belonging to the head/anchor foot node of a syntagma (this node specifies the eligible part(s) of speech, among other things). In case of successful unification, the lemma triggers the matching syntagma. The allocation of input lemmas to LSI positions is controlled by a router that is located at the exit port of the Mental Lexicon and, confronted with a new lemma, assigns it to the next “empty” LSI position. Input words that are ambiguous with respect to word class go to the same array position (but will often bind different syntagma types).¹⁵ When the system embarks on a new sentence, the router copies the feature matrix of the first arriving lemma to the seven cells at the leftmost array position. This triggers seven parallel unification attempts, of which only one will succeed (or a few, in case of word class ambiguity). The second input lemma encountered by the router goes to the second set of seven unifiers, etc.

How do lemmas get hold of their referential indices?

During *decoding* episodes, the incoming lemmas recognized in the spoken or written input stream transmit activation to concepts, and from concepts to the mental models they designate (both presumably residing in neocortical brain regions). Within the Hippocampal Complex, the active mental models—or, rather, active *copies* of mental models—will then be tagged with referential indices (contextual coordinates), which, I assume, are immediately copied into the registers of the input lemmas currently occupying the LSI, and stay there for the duration of the decoding episode. *Encoding* episodes often start with a mental scene imagined by the speaker, who then looks up a set of concepts covering the mental models and their interrelationships. The referential indices assigned to the

¹⁵ Lemmas associated with the same lexically ambiguous input word (e.g., with the noun/verb *can*) compete with one another via prewired, mutually inhibitory links located in the Mental Lexicon (ML). The competitor that is more successful as a unification partner in U-Space, will feed more activation back to its ML entry (e.g., the verb *can*) than the less fortunate counterpart will be able to send to *its* entry (the noun *can*). As a consequence, the latter will extinguish, and the syntagma it originally managed to bind (if any), will soon decay and become available as a binding partner for another lemma.

mental models get copied into the conceptual dependency graph covering the scene, and thence into the annotations of the lemmas that enter the LSI. There, the indices and their values have a crucial influence on the emerging root-to-foot bindings: Because the unifiers in LSI cells take index features into consideration explicitly, the index values of root and foot should be identical, as discussed above.¹⁶

How Does the U-Space Communicate its Output to Other Processing Modules?

The results of U-Space processing get recorded in the feature matrices of the lemmas occupying LSI cells. However, U-Space activity leaves the lemma-to-position assignments in the LSI intact. This applies to both coding modalities. For decoding tasks, this is not unexpected since the input lemma sequence mirrors the input word string; and once the U-Space has delivered its functional dependency graph, the linear order of the input words plays no role anymore in the conversions of functional to conceptual graphs, and from the latter to mental scenes. For encoding tasks, however, the fixed LSI positions create a problem because they mirror the temporal orders in which imagery and conceptual processes construct and deliver the mental models and concepts—orders that are not informed by grammatical processes, hence risking to deviate from the linear orders allowed by the topologies in the recruited syntagmas.

This problem is solved by the RTN’s *readout* function. When the RTN is traversing a syntagma hierarchy and descends into the lexical Head foot node of some syntagma, it accesses the binding lemma that resides in the corresponding LSI cell. Activating this cell causes a copy of the feature matrix of the lemma to be transmitted to a special *Morpho-syntactic Buffer*. This buffer belongs to the neuronal mechanism subserving the *Phonological Encoding* stage of sentence production. In the psycholinguistic literature on language production, it is generally assumed that the stage of Grammatically Encoding is followed by a stage of Phonological Encoding (e.g., Garrett 1975; Kempen and Hoenkamp 1982, 1987; Levelt 1989). Here, lemmas are converted to lexemes with inflectional properties that match the morpho-syntactic features computed and attached by the Unification Space. Phonological encoding also includes prosodic processing (yielding intonation contours), and may give rise to contractions (e.g., *it is* → *it’s*; *going to* → *gonna*). The resulting prosodically marked string of phonemes is finally passed on to phonetic modules for overt articulation. Acheson et al. (2011) present fMRI and rTMS evidence for the left posterior superior temporal gyrus (lpSTG) as subserving the

¹⁶ To prevent a misunderstanding, unifiers that check agreement relations between sister constituents (e.g. Subject-verb agreement) do not take index values into account. See also the final paragraphs of Section “[Microcircuitry for feature unification.](#)”

phonological encoding stage of language production. This area belongs to the temporo-parietal regions that are “a strong candidate for the neural substrate of storage, both in sentence processing and item-based tasks” (Meyer et al. 2012; see also Buchsbaum and D’Esposito 2008, for discussion).

A router supervising access to the Morpho-syntactic Buffer appends a copy of every new lemma (more precisely, a copy of its feature matrix) to the list of matrices currently residing in the buffer. This procedure builds incrementally a grammatically well-formed list of annotated lemmas that contains all information needed to convert the list of lemmas to a list of word forms (lexemes). Actually, it is not necessary to send to the buffer complete copies of the annotated lemmas in LSI cells: The morpho-syntactic features that select the appropriate word forms suffice (e.g., number, person, and tense of to-be-realized finite verbs)—thence the name “Morpho-syntactic” buffer.

As part of the language *comprehension* process, words recognized in the spoken or written input must be converted to lemmas with morpho-syntactic features. It seems reasonable to assume that this lexeme-to-lemma conversion task is carried out by the same module that performs lemma-to-lexeme conversion during phonological encoding. If this assumption is correct, the Morpho-syntactic buffer has double functionality: In comprehension tasks, it temporarily stores the lemma input to the LSI; in production tasks, it receives the output from the LSI. This suggests a mechanism to check whether, during grammatical decoding, the linear order entailed by the current parse dovetails with the actual input word order and, if not, to block the parse. Remember from the discussion of the toy example *Dogs cats bite fight* (e.g., Fig. 2d) the danger for the two verbs to link up to the two NPs in the wrong order. The correctness of such bindings can be verified, in principle, if the readout function is running not only during encoding but also during decoding—thus, in a sense, shadowing the actual input. Each time a new lemma is appended to the Morpho-syntactic buffer, the phonological processor can check whether it matches with the next input lemma in the list compiled at input.

A similar shadowing mechanism could be operative in grammatical encoding, but here the shadowed lemmas belong to the functional dependency graph delivered by the U-Space. This speculative idea presupposes the existence of what might be called a *Functional dependency Buffer* that short-term-stores lemmas containing all information needed to convert functional to conceptual dependency graphs, and back. Such a buffer would enable quick comparisons between the meaning intended by the speaker, and the meaning returned by the shadowing function. In case of a mismatch, a U-Space error must have occurred—an overt or a covert speech error. This second type of shadowing could take place, at least insofar as the online construction of mental images is involved, within the Hippocampal Complex. Kumaran and Maguire (2007) argue that the hippocampus includes a comparator mechanism enabling rapid detection and signaling of novelty in perceived

scenes (see also Duncan et al. 2012; Poppenk et al. 2008). Both types of shadowing could provide powerful safeguards against U-Space errors in grammatical coding. However, whether they are actually applied, and under what kind of conditions, is a topic awaiting further investigation.

Where is the Neural Substrate of the Unification Space?

The present section has made clear that grammatical coding tasks encompass more than just the tasks fulfilled by the Unification Space. While the U-Space is primarily responsible for the unification and the linearization of annotated lemmas, additional processing is required for—among other things—(1) lemma selection, (2) assembly of functional dependency graphs, (3) assembly of conceptual dependency graphs, and (4) referential processing. It is too early to make specific proposals about how these subtasks are compartmentalized into modules and distributed over brain regions. However, it is hardly controversial that the neural infrastructure in their service is located in or close to the left superior/medial temporal lobe (Pettersson and Hagoort 2012), including the Hippocampal Complex. It is important to realize that these subtasks are by no means trivial or marginal. For instance, in *encoding* mode, they include the selection of all lemmas for the sentence under construction, which in turn includes the selection of pronouns and other closed-class vocabulary (“syntactic words”). Furthermore, the assembly of the functional dependency graph (in the form of a set of annotated lemmas) fully determines the hierarchy of syntagmas to be recruited in U-Space (through the referential indices). In *decoding* mode, the subtasks are executed in reverse direction: converting functional to conceptual dependency graphs, lemmas to concepts (which often necessitates dealing with lexical ambiguity, and includes pronoun resolution), and concepts to mental models; finally, the models are combined into integrated iconic mental scenes in accordance with conceptual relations and referential indices.

The division of labor between subtasks performed by the Unification Space on the one hand and those that are localized to the left superior/medial temporal lobe on the other, raises the question of which brain area(s) subserve(s) the Unification Space. The traditional answer, which is based on a plethora of empirical evidence, points to Broca’s area (the left inferior frontal gyrus (IIFG) and surrounding areas; details need not concern us here). However, much of that evidence is equivocal and open to alternative theoretical interpretations. One such alternative holds that Broca’s area is a domain-general processor that subserves neural computations of a special type—variously characterized as (1) conflict resolution, (2) sequential processing, or (3) syntactic working memory—and that, in this capacity, it is a crucial ingredient of grammatical coding. Irrespective of whether Broca’s area is essential for other cognitive functions, I observe that U-Space function as elaborated in this paper is

based on (1) conflict resolution (competition via lateral inhibition) and (2) engenders sequential structures; furthermore, the syntagmas and their sustained activity during coding episodes fulfills the role of (3) working memory. Hence, I see no empirical objections against assigning some grammatical coding subtasks to Broca's area and the IIFG, in particular those subtasks dealing with morpho-syntactic unification and syntactic linearization. However, note that other subtasks of grammatical coding—particularly its lexical, conceptual, and referential aspects—are most probably to be localized to superior/medial temporal areas.¹⁷

Interregional Connections

Recently, Bornkessel-Schlesewsky and Schlewsky (2013:67; see also Alday et al. 2013) proposed to localize *all* linguistic structure-building operations to temporal (and parietal) regions: “*Frontal cortex does not subservise linguistic processing functions*” but “[...] subserves control functions only” (italics theirs).¹⁸ This proposal is an integral part of, and largely motivated by, their more encompassing theoretical framework on the function of the ventral and the dorsal processing streams

¹⁷ For two fMRI studies whose data are in unison with the assumption of bilocation of grammatical decoding subtasks, see Snijders et al. (2009) and Pallier et al. (2011), with Dutch and French, respectively, as target languages.

¹⁸ Bornkessel-Schlesewsky and Schlewsky (2013:70) cite the two data sets mentioned in the preceding footnote, interpreting them in favor of their alternative framework. There, they also cite a sentence production experiment by Ye et al. (2011) as evidence that “posterior temporal regions may be involved in the linearisation of elements, i.e. in determining the linear order of syntactic constituents”. In this experiment, German-speaking participants expressed a sequence of two events (each suggested by a drawing) in the form of sentences that always started with an adverbial clause introduced by the subordinating conjunction *before* or *after* (e.g. the German translation equivalents of *Before I open the mail, I sit down* or *After I sit down, I open the mail*). The subordinating conjunction to be used in a given trial was indicated by a color cue presented immediately after the drawings.

It is well-known from behavioral studies that *before* sentences of this type are harder to produce than *after* sentences, presumably because, with *before*, the events cannot be mentioned in their chronological order. This extra difficulty was mirrored by higher levels of activation in superior temporal regions. According to the researchers and to Bornkessel-Schlesewsky et al., this finding suggests that temporal regions are involved in linear order computations for upcoming sentences. Although this interpretation is valid in itself, it is not the only one possible. The extra activation recruited by *before* sentences could very well be a consequence of the need to inhibit the tendency to select the more natural concept/lemma underlying *after*—arguably more “natural” concept/lemma here, since in trials with *after* the adverbial clause was produced first and expressed the first event. If so, some other neural structure must have been in charge of linearization. Indeed, in addition to the posterior temporal region, certain frontal regions were also more active during *before* trials, e.g. the left middle frontal gyrus. Hence, this area may have been the one responsible for the linearization aspects of the task, by controlling the order of subordinate and main clause as imposed by the experimenter.

that connect the frontal and the temporal regions involved in sentence comprehension. The dorsal stream runs between the IIFG and posterior regions of the left temporal lobe (including the superior temporal gyrus and the superior temporal sulcus, pSTG/pSTS). The ventral stream connects the IIFG with the medial temporal lobe (MTL). Although Bornkessel-Schlesewsky et al.'s position on the role of the frontal cortex clearly deviates from mine, another aspect of their theoretical framework does fit the present model. Bornkessel-Schlesewsky et al. present strong empirical and theoretical arguments for a distinction between “time-dependent” and “time-independent” aspects of sentence processing, and assign the former to the dorsal stream, the latter to the ventral stream. The specific subtasks they characterize as time-dependent and -independent follow from their eADM model of sentence comprehension (Bornkessel-Schlesewsky and Schlewsky 2009).

The present model suggests another characterization of time-dependent vs. time-independent sentence processing: Time-dependent processing deals with linearly ordered structures, time-independent processing with unordered dependency structures. Strings of annotated lemmas entering U-Space during sentence comprehension are time-dependent, as are annotated lemma strings that U-Space delivers as output during sentence production. During (auditory) comprehension episodes, the lemma strings originate in the left posterior STS/STG (cf. Okada and Hickock 2006), presumably selected during conversion of phonological wordforms to lemmas with morpho-syntactic annotations (number, person, tense, case, etc.) Strings of lemmas thus annotated activate positions in the lemma-syntactic interface (LSI) of the U-Space and start up the grammatical decoding process. During sentence production, U-Space assigns linear order to input lemmas belonging to a referentially annotated functional dependency graph, and the RTN then transmits them one-by-one to the left pSTG/pSTS for Phonological Encoding (cf. Acheson, et al. 2011), that is, via the dorsal stream. Time-independent processing as part of sentence comprehension concerns the output of the U-Space, i.e. the functional-dependency structure which is converted to a conceptual-dependency structure in MTL and referentially processed in HC: the ventral stream. During sentence production episodes, the ventral stream assembles the referential, conceptual, and functional structures for the upcoming sentence, and transmits it to the U-Space in the IIFG.

This architecture affords two important advantages if we assume that all interregional connections between the four neural centers (left IFG, left MTL, left pSTG/pSTS, and HC) are bidirectional. First, it enables quick feedback loops in the service of self-monitoring. For instance, during grammatical *decoding*, the system can use the dorsal stream to check whether the word order entailed by the parse (and read out by the RTN) is faithful to the input word order—recall the idea of a morpho-syntactic buffer proposed earlier on in the present section. Such a buffer is presumably located in the

left pSTG/pSTS. In case of a mismatch, the buffer can immediately transmit inhibition back to the offending unification partners in IIFG. The ventral stream can improve the quality (plausibility, veridicality) of a parse in a similar manner by inhibiting low-quality unifications, thereby possibly giving defeated unifications a new chance. (This presupposes, of course, that the grammatical coding mechanisms in the medial temporal lobe have direct access to knowledge in long-term memory.) The decoding system thus becomes “constraint-based” in the psycholinguistic sense of allowing conceptual and pragmatic factors to affect syntactic decisions at the earliest possible moment, without a delay due to a preceding, purely syntactic processing stage. Self-monitoring during grammatical *encoding* is enabled, for instance, by temporarily “clamping” the set of annotated lemmas residing in the LSI at the end of an encoding episode, and feeding these lemmas back into the MTL mechanism responsible for assemblage of functional and conceptual dependency structures. This may reveal differences between the speaker’s original communicative intention and the intention that was actually conveyed. The second advantage of the split between ventral and dorsal streams is that it facilitates parallel processing by the neural components involved in computing the linear aspects of sentence structure on the one hand, and the dependential aspects on the other—a prerequisite for constraint-based grammatical coding.

Concluding Remarks

The present study sought to answer two questions. First, how is syntactic structure formation of the complexity required by natural-language grammars possible in a fixed, preexisting neural network without the need for online creation of new connections or associations (without “rapid learning”)? Second, is it realistic to assume that the seemingly disparate instantiations of syntactic structure formation in grammatical encoding vs. grammatical decoding can run on the same neural infrastructure? The Unification-Space design put forward in this paper suggests a positive answer to both questions: It provides the blueprint for a syntactic structure formation mechanism that is entirely based on prewired circuitry (except for aspects of referential processing), and can subserve decoding as well as encoding tasks (insofar as these tasks rely on the U-Space). However, many questions remain unanswered. The following—somewhat arbitrary—list of open issues concludes the paper.

1. As the first word of the title suggests, the system described in the preceding pages is incomplete in many respects. In particular, a detailed description is missing of the dynamics of the competition-based optimization process that underlies the (re-)construction of syntactic trees. An implementation of such an optimization process has appeared in the SINUS papers by Vosse and Kempen (2008, 2009a), but the syntactic structures targeted by the new model are much more complex. Also lacking is the design of a component that converts conceptual dependency graphs into functional dependency graphs, and vice-versa. This is an important part of grammatical coding the model leaves open. Given this state of affairs, it is too early to expect detailed predictions about psycholinguistic or neurobiological properties of grammatical coding—predictions that are novel (i.e., not built into the model a priori) and contrastive (enabling to pit the model against alternative theoretical frameworks). This, of course, is not a license to unbridled theoretical speculation but an urgent exhortation to implement the model and run crucial computer simulations.
2. Nonetheless, although the proposed neuroarchitecture needs detailed elaboration and computational implementation, already in its current form it can shed light on the theoretical interpretation of some extant empirical data. A long-standing controversy is over serial syntax-first theories of grammatical decoding (e.g. Friederici 2002, 2011) versus interactive constraint-based theories (e.g. Hagoort 2005). In essence, the present model belongs to the latter category, but its behavior may give the impression of a decoding process organized in stages. Strong phrase-structure violations in the incoming sentence, e.g. substitution of a strongly expected noun by a verb, will be detected quickly when the unexpected input word does not allow the RTN to jump to a legal topological position where the new input word can attach. Although conceptual and referential processing is initiated in parallel with syntactic decoding in U-Space, it will often take more time, thereby suggesting a second, semantic/interpretive stage. The dynamic optimization process will always attempt to arrive at a structure that meets all prevailing syntactic, conceptual and referential constraints. The time it takes for the competition to settle down will increase when input properties are not entirely compatible and prevent a fully satisfactory solution. This gives the impression of a third processing stage devoted to reanalysis and repair. Thus, a strictly interactive constraint-satisfaction model can create the impression of operating in a serial syntax-first manner; however, the stages are an emergent, not a design property.

A related debate concerns the characterization of syntax-related ERP effects such as ELAN (Early Left Anterior Negativity) and P600. The ELAN response may be causally related to an RTN blockade—an input word preventing the RTN to proceed to the next position due to a wrong PoS (word category) value. The P600 may be viewed as the system’s reaction to an input word that, although allowing the RTN to proceed, yields a suboptimal binding that the RTN accepts nevertheless,

but only after coercion. The reason for the suboptimal quality of the binding may be morpho-syntactic or conceptual (e.g., implausibility). This tentative proposal, I surmise, fits under the umbrella of a recent hypothesis proposed by Brouwer et al. (2012:138): “The P600 component is a family of late positivities that reflect the word-by-word construction, reorganization, or updating of a mental representation of what is being communicated.” In the model, these activities take the form of a unification process that is somewhat delayed due to mismatching features but succeeds nevertheless.

3. Even if an implementation of the model can be built that encodes and decodes sentences of the required complexity level (generated by mildly context-sensitive grammars), there is no guarantee that a scaled-up version embodying a substantial grammar fragment of a language and incorporating a realistic-sized vocabulary will work. (See Stewart and Eliasmith (2012) on scalability problems posed by some extant neurocomputational models of language processing.) The number of prewired connections needed to transport lexical information within the Mental Lexicon (ML) and between the ML and the Unification Space may be forbidding. The ML of adult language users contains at least 50,000 items (a conservative estimate that does not include proper names, multiword expressions, and several other types of items). This suggests a full-blown system contains no less than 50,000 prewired bidirectional temporal-frontal connections capable of transporting registers that each represent pretty complex feature matrices.

However, two considerations may temper pessimism. First, the ML includes many groups of lemmas whose members have the same or similar feature matrices. Suppose that lexical entries are organized as inheritance (or specialization) hierarchies. For example, ditransitive verbs share all their features with monotransitive verbs, except for the optional Indirect Object. Monotransitives may be viewed as a subclass (specialization) of the intransitives, distinguished only by the addition of a Direct Object. Inheritance hierarchies enable elimination of great quantities of redundant information. They can be realized in neural nets by introducing nodes that do not represent a specific lemma but a feature matrix, and connect lower, more specific nodes with higher, more general nodes via copy links. Individual lemma constitute the lowest level of such a hierarchy; when activated, they copy their “private” feature matrix to the next higher node, where it is unified with the matrix of the latter, etc., until a complete feature matrix arrives at the exit port of the ML, ready to be dispatched to frontal regions. A hierarchically organized ML with inheritance also reduces, to a considerable extent, the problem of processing novel words (e.g., see Feldman 2013). When a new word is

encountered, it only needs to learn its place in the lexical hierarchy, close to already known, grammatically similar words. This learning task arguably is much easier than learning all grammatical properties of the new word from scratch.

Second, although the ML probably allows activation of multiple lexical entries in parallel, its entrance and exit ports may be controlled by a router that enforces sequential admission and discharge. Many aspects of language comprehension and production proceed incrementally, sequentially. Therefore, the enormous size of the ML, in itself, does not necessitate high-capacity temporal-frontal pathways.

4. The assumption that the system works with a fixed number of exemplars of each syntagma type will prompt the criticism that it imposes a hard upper limit on sentence length and does not yield graceful degradation. However, graceful degradation results primarily from the system’s dynamical properties, e.g. from activation decay preventing late constituents of a long sentence to get access to, and take account of, properties of early constituents. Such effects may make themselves felt already before hard capacity limits are reached. Moreover, one can add special measures that attenuate the upper limit on sentence length—although at the risk of introducing errors—such as “recycling” syntagmas that were recruited early on during a given sentence coding episode but whose activation level has since dropped close to or below threshold.
5. An important research topic concerns the child’s acquisition of a system such as the one proposed here. Hopefully, the present portrayal of how the brain divides the grammatical coding faculty into less complex components, stimulates a divide-and-conquer strategy, reminiscent of Elman’s (1993) adage “the importance of starting small”: Focus on techniques that can learn, sequentially or in parallel, the tasks performed by separate model components (e.g., individual syntagmas and RTNs, individual unification circuits) rather than on techniques that target encoding/decoding tasks in their full complexity (e.g., RNNs that, from the outset of training, are confronted with the entire set of input–output relations defined by a nontrivial recursive grammar). The model perhaps facilitates the divide-and-conquer research strategy by carving nature at its joints.

Information Sharing Statement

Given that the paper describes the *design* of a neurocomputational model, we cannot share program code of a computer implementation, or supporting experimental data collected at our institute. Readers interested in implemented

versions of *predecessors* of the current model can download three demonstration movies (in MOV format) corresponding to the syntactic parsing models described in Kempen and Vosse (1989), Vosse and Kempen (2000) and Vosse and Kempen (2009a). The movies can be downloaded from http://www.gerardkempen.nl/UnificationSpace_parser_demos.html/. The most recent movie concerns the SINUS model described in Vosse and Kempen (2008, 2009a).

A demonstration version of the SINUS model, implemented in C++ by Theo Vosse, runs as a PowerPC application on Mac computers under OS X (v6.0, Snow Leopard, or lower). The system, which parses simple English and German sentences, is available from the author (gerard.kempen@mpi.nl) upon request.

Acknowledgments I am very grateful to Karin Harbusch, Peter Hagoort, Karl-Magnus Petersson, the editor of this special issue, and three anonymous reviewers for extensive comments on earlier versions of the paper. Of course, the responsibility for the final content is solely mine.

References

- Acheson, D. J., & MacDonald, M. C. (2009). Verbal working memory and language production: common approaches to the serial ordering of verbal information. *Psychological Bulletin*, *135*, 50–68.
- Acheson, D. J., Hamidi, M., Binder, J. R., & Postle, B. R. (2011). A common neural substrate for language production and verbal working memory. *Journal of Cognitive Neuroscience*, *23*, 1358–1367.
- Alday, P.M., Schlesewsky, M., & Bornkessel-Schlesewsky, I. (2013). Towards a computational model of actor-based language comprehension. *Neuroinformatics*. doi:10.1007/s12021-013-9198-x (in this issue).
- Arbib, M. A. (1989). *The Metaphorical Brain 2: Neural Networks and Beyond*. New York: Wiley.
- Baddeley, A. D. (2012). Working memory: theories, models, and controversies. *Annual Review of Psychology*, *12*, 1–29.
- Baggio, G., & Hagoort, P. (2011). The balance between memory and unification in semantics: a dynamic account of the N400. *Language & Cognitive Processes*, *26*, 1338–1367.
- Barrès, V., & Lee, J. (2013). Template Construction Grammar: from visual scene description to language comprehension and agrammatism. *Neuroinformatics*. doi:10.1007/s12021-013-9197-y (in this issue).
- Barsalou, L. W. (1999). Perceptual symbol systems. *The Behavioral and Brain Sciences*, *22*, 577–610.
- beim Graben, P., Gerth, S., & Vasisth, S. (2008). Towards dynamical system models of language-related brain potentials. *Cognitive Neurodynamics*, *2*, 229–255.
- Bird, C. M., & Burgess, N. (2008). The hippocampus and memory: insights from spatial processing. *Nature Reviews Neuroscience*, *9*, 182–194.
- Bornkessel-Schlesewsky, I., & Schlesewsky, M. (2009). The role of prominence information in the real time comprehension of transitive constructions: A cross-linguistic approach. *Language and Linguistics Compass*, *3*, 19–58.
- Bornkessel-Schlesewsky, I., & Schlesewsky, M. (2013). Reconciling time, space and function: a new dorsal–ventral stream model of sentence comprehension. *Brain and Language*, *125*, 60–76.
- Brouwer, H., Fitz, H., & Hoeks, J. (2012). Getting real about semantic illusions: rethinking the functional role of the P600 in language comprehension. *Brain Research*, *1446*, 127–143.
- Buchsbaum, B. R., & D'Esposito, M. (2008). The search for the phonological store: from loop to convolution. *Journal of Cognitive Neuroscience*, *20*, 762–778.
- Byrne, P., Becker, S., & Burgess, N. (2007). Remembering the past and imagining the future: a neural model of spatial memory and imagery. *Psychological Review*, *114*, 340–375.
- Cowan, N. (2005). *Working memory capacity*. Hove, UK: Psychology Press.
- De Smedt, K.J.M. (1990). *Incremental sentence generation: A computer model of grammatical encoding*. PhD dissertation, Radboud University, Nijmegen.
- Diana, R. A., Yonelinas, A. P., & Ranganath, C. (2007). Imaging recollection and familiarity in the medial temporal lobe: a three-component model. *Trends in Cognitive Sciences*, *11*, 379–386.
- Duff, M. C., & Brown-Schmidt, S. (2012). The hippocampus and the flexible use and processing of language. *Frontiers in Human Neuroscience*, *6*, 1–11.
- Duff, M. C., Gupta, R., Hengst, J., Tranel, D., & Cohen, N. J. (2011). The use of definite references signals declarative memory: evidence from Hippocampal amnesia. *Psychological Science*, *22*, 666–673.
- Duncan, K., Ketz, N., Inati, S. J., & Davachi, L. (2012). Evidence for area CA1 as a match/mismatch detector: a high-resolution fMRI study of the human hippocampus. *Hippocampus*, *22*, 389–398.
- Eberhard, K. M., Cutting, J. C., & Bock, J. K. (2005). Making syntax of sense: number agreement in sentence production. *Psychological Review*, *112*, 531–559.
- Elman, J. L. (1991a). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, *7*, 195–225.
- Elman, J. L. (1991b). Finding structure in time. *Cognitive Science*, *14*, 179–211.
- Elman, J. L. (1993). Learning and development in neural networks: the importance of starting small. *Cognition*, *48*, 71–99.
- Featherston, S. (2004). Bridge verbs and V2 verbs—the same thing in spades? *Zeitschrift für Sprachwissenschaft*, *23*, 181–209.
- Feldman, J. (2013). The neural binding problem(s). *Cognitive Neurodynamics*, *7*, 1–11.
- Ferreira, F., & Clifton, C., Jr. (1986). The independence of syntactic processing. *Journal of Memory and Language*, *25*, 348–368.
- Friederici, A. D. (2002). Towards a neural basis of auditory sentence processing. *Trends in Cognitive Science*, *6*, 78–84.
- Friederici, A. D. (2011). The brain basis of language processing. *Physiological Reviews*, *91*, 1357–1392.
- Friederici, A. D. (2012). The cortical language circuit: from auditory perception to sentence comprehension. *Trends in Cognitive Sciences*, *16*, 262–268.
- Garrett, M. (1975). The analysis of sentence production. In G. Bower (Ed.), *The psychology of learning and motivation* (Vol. 9). New York: Academic.
- Gayler, R. W. (2003). Vector symbolic architectures answer Jackendoff's challenges for cognitive neuroscience. In Slezak, P.P. (Ed.), *Proceedings of the Joint International Conference on Cognitive Science*. University of New South Wales. Available at <http://cogprints.org/3983/>
- Gerth, S., & beim Graben. (2009). Unifying syntactic theory and sentence processing difficulty through a connectionist minimalist parser. *Cognitive Neurodynamics*, *3*, 297–316.
- Gibson, E., & Warren, T. (2004). Reading-time evidence for intermediate linguistic structure in long-distance dependencies. *Syntax*, *7*, 55–78.

- Glenberg, A., Meyer, M., & Lindem, K. (1987). Mental models contribute to foregrounding during text comprehension. *Journal of Memory and Language*, 26, 69–83.
- Gupta, P., & MacWhinney, B. (1997). Vocabulary acquisition and verbal short-term memory: computational and neural bases. *Brain and Language*, 59, 267–333.
- Hagoort, P. (2003). How the brain solves the binding problem for language: a neurocomputational model of syntactic processing. *NeuroImage*, 20, S18–S29.
- Hagoort, P. (2005). On Broca, brain, and binding: a new framework. *Trends in Cognitive Sciences*, 9, 416–423.
- Hagoort, P. (2007). The memory, unification, and control (MUC) model of language. In A. Meyer, L. Wheeldon, & A. Krott (Eds.), *Automaticity and control in language processing*. Hove UK: Psychology Press.
- Harbusch, K., & Kempen, G. (2002). *A quantitative model of word order and movement in English, Dutch and German complement constructions*. *Proceedings of the 19th International Conference on Computational Linguistics (COLING-2002), Taipei (Taiwan)*. San Francisco: Morgan Kaufmann.
- Harnad, (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42, 335–346.
- Henderson, J. (1994). Connectionist syntactic parsing using temporal variable binding. *Journal of Psycholinguistic Research*, 23, 353–379.
- Hinaut, X., & Dominey, P. F. (2013). Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing. *PLoS One*, 8, e52946.
- Howard, M. D., Bhattacharyya, R., O'Reilly, R. C., Ascoli, G., & Fellous, J.-M. (2011). Adaptive recall in hippocampus. In A. V. Samsonovich & K. R. Jóhannsdóttir (Eds.), *Biologically inspired cognitive architectures 2011*. Amsterdam/Washington: IOS Press.
- Hummel, J. E., & Holyoak, K. J. (2003). A symbolic-connectionist theory of relational inference and generalization. *Psychological Review*, 110, 220–264.
- January, D., Trueswell, J. C., & Thompson-Schill, S. L. (2009). Colocalization of Stroop and syntactic ambiguity resolution in Broca's area: Implications for the neural basis of sentence processing. *Journal of Cognitive Neuroscience*, 21, 2434–2444.
- Johnson-Laird, P. N. (1983). *Mental models: toward a cognitive science of language, inference and consciousness*. Cambridge: Cambridge University Press.
- Johnson-Laird, P. N. (2006). *How we reason*. Oxford: Oxford University Press.
- Joshi, A. K. (1985). Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions? In D. Dowty, L. Karttunen, & A. M. Zwicky (Eds.), *Natural language parsing: Psychological, computational and theoretical perspectives*. New York: Cambridge University Press.
- Kanerva, P. (2009). Hyperdimensional computing: an introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computing*, 1, 139–159.
- Kaplan, R. M. (1972). Augmented transition networks as psychological models of sentence comprehension. *Artificial Intelligence*, 3, 77–100.
- Kempen, G. (2000). Could grammatical encoding and grammatical decoding be subserved by the same processing module? *The Behavioral and Brain Sciences*, 23, 38–39.
- Kempen, G., & Harbusch, K. (2002). Performance Grammar: A declarative definition. In A. Nijholt, M. Theune, & H. Hondorp (Eds.), *Computational Linguistics in the Netherlands 2001*. Amsterdam: Rodopi.
- Kempen, G., & Harbusch, K. (2003). Dutch and German verb constructions in Performance Grammar. In P. A. M. Seuren & G. Kempen (Eds.), *Verb constructions in German and Dutch*. Amsterdam: Benjamins.
- Kempen, G., & Hoenkamp, E. (1982). Incremental sentence generation: implications for the structure of a syntactic processor. In J. Horecky (Ed.), *Proceedings of the Ninth International Conference on Computational Linguistics (COLING 1982), Prague, July 1982*. Amsterdam: North-Holland.
- Kempen, G., & Hoenkamp, E. (1987). An incremental procedural grammar for sentence formulation. *Cognitive Science*, 11, 201–258.
- Kempen, G., & Vosse, T. (1989). Incremental syntactic tree formation in human sentence processing: a cognitive architecture based on activation decay and simulated annealing. *Connection Science*, 1, 273–290.
- Kempen, G., & Vosse, T. (1994). The Unification Space: A hybrid model of human syntactic processing. In *The Seventh Annual CUNY Conference on Human Sentence processing*. New York NY: CUNY Graduate Center.
- Kempen, G., Olsthoorn, N., & Sprenger, S. A. (2012). Grammatical workspace sharing during language production and language comprehension: evidence from grammatical multitasking. *Language & Cognitive Processes*, 27, 345–380.
- Kumaran, D., & Maguire, E. A. (2007). Which computational mechanisms operate in the hippocampus during novelty detection? *Hippocampus*, 17, 735–748.
- Kumaran, D., & McClelland, J. L. (2012). Generalization through the recurrent interaction of episodic memories: a model of the hippocampal system. *Psychological Review*, 119, 573–616.
- Levelt, W. J. M. (1989). *Speaking: From intention to articulation*. Cambridge MA: MIT Press.
- Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29, 375–419.
- MacKay, D. G., Johnson, L. W., Fazel, W., & James, L. E. (2013a). Compensating for language deficits in amnesia I: H.M.'s spared retrieval categories. *Brain Sciences*, 3, 262–293.
- MacKay, D. G., Johnson, L. W., & Hadley, C. (2013b). Compensating for language deficits in amnesia II: H.M.'s spared versus impaired encoding categories. *Brain Sciences*, 3, 415–459.
- Marcus, G. F. (2001). *The algebraic mind*. Cambridge MA: MIT Press.
- Marcus. (2009). How does the mind work? Insights from Biology. *Topics in Cognitive Science*, 1, 145–172.
- Marcus, G. F. (2013). Evolution, memory, and the nature of syntactic representation. In J. J. Bolhuis & M. Everaert (Eds.), *Birdsong, speech, and language: Exploring the evolution of mind and brain*. Cambridge MA: MIT Press.
- McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 95, 419–457.
- Menenti, L., Gierhan, S. M. E., Segaert, K., & Hagoort, P. (2011). Shared language: Overlap and segregation of the neuronal infrastructure for speaking and listening revealed by functional MRI. *Psychological Science*, 22, 1173–1182.
- Meyer, L., Obleser, J., Anwander, A., & Friederici, A. D. (2012). Linking ordering in Broca's area to storage in left temporoparietal regions: the case of sentence processing. *NeuroImage*, 62, 1987–1998.
- Novick, J. M., Trueswell, J. C., & Thompson-Schill, S. L. (2005). Cognitive control and parsing: reexamining the role of Broca's area in sentence comprehension. *Cognitive, Affective, & Behavioral Neuroscience*, 5, 263–281.
- Novick, J. M., Trueswell, J. C., & Thompson-Schill, S. L. (2010). Broca's Area and language processing: evidence for the cognitive control connection. *Language and Linguistic Compass*, 4, 906–924.
- Nozari, N., Dell, G. S., & Schwarz, M. F. (2011). Is comprehension necessary for error detection? A conflict-based account of monitoring in speech production. *Cognitive Psychology*, 63, 1–33.

- O'Reilly, R. C., Bhattacharyya, R., Howard, M. D., & Ketz, N. (2011). Complementary Learning Systems. *Cognitive Science*, 35, 1–20.
- Okada, K., & Hickock, G. (2006). Left posterior auditory-related cortices participate both in speech perception and speech production: neural overlap revealed by fMRI. *Brain and Language*, 98, 112–117.
- Pallier, C., Devauchelle, A.-D., & Dehaene, S. (2011). Cortical representations of the constituent structure of sentences. *Proceedings of the National Academy of Sciences of the United States of America*, 108, 2522–2527.
- Perham, N., Marsh, J. E., & Jones, D. M. (2009). Syntax and serial recall: How language supports short-term memory for order. *Quarterly Journal of Experimental Psychology*, 62, 1285–1293.
- Petersson, K.-M., & Hagoort, P. (2012). The neurobiology of syntax: beyond string sets. *Philosophical Transactions of the Royal Society B*, 367, 1971–1983.
- Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence*, 46, 77–105.
- Poppenk, J., Walia, G., McIntosh, A. R., Joanisse, M. F., Klein, D., & Köhler, S. (2008). Why is the meaning of a sentence better remembered than its form? An fMRI study on the role of novelty-encoding processes. *Hippocampus*, 18, 909–918.
- Pulvermüller, F. (2010). Brain embodiment of syntax and grammar: discrete combinatorial mechanisms spelt out in neuronal circuits. *Brain and Language*, 112, 167–179.
- Sachs, J. S. (1967). Recognition memory for syntactic and semantic aspects of connected discourse. *Perception & Psychophysics*, 2, 437–442.
- Segaert, K., Menenti, L., Weber, K., Petersson, K. M., & Hagoort, P. (2012). Shared syntax in language production and language comprehension: an fMRI study. *Cerebral Cortex*, 22, 1662–1670.
- Shastri, L., & Ajjanagadde, V. (1993). From simple associations to systematic reasoning: a connectionist representation of rules, variables, and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences*, 16, 417–494.
- Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46, 159–216.
- Snijders, T. M., Vosse, T., Kempen, G., van Berkum, J. J. A., Peterson, K.-M., & Hagoort, P. (2009). Retrieval and unification of syntactic structure in sentence comprehension: an fMRI study using word category ambiguity. *Cerebral Cortex*, 19, 1493–1503.
- Staub, A. (2010). Eye movements and processing difficulty in object relative clauses. *Cognition*, 116, 71–86.
- Stevenson, S. (1993). A competition-based explanation of syntactic attachment preferences and garden path phenomena. In *Thirty-first Annual Meeting of the Association for Computational Linguistics*. Columbus OH: Ohio State University.
- Stewart, T., & EliaSmith, C. (2012). Compositionality and biologically plausible models. In M. Werning, W. Hinzen, & E. Machery (Eds.), *The Oxford handbook of compositionality*. Oxford UK: Oxford University Press.
- Thothathiri, M., Kim, A., Trueswell, J. C., & Thompson-Schill, S. L. (2012). Parametric effects of syntactic–semantic conflict in Broca's area during sentence processing. *Brain and Language*, 120, 259–264.
- Tooley, K., & Bock, J. K. (2011). *The equivalence of syntactic priming in comprehension and production*. *Proceedings of the 24th Annual CUNY Conference on Human Sentence Processing*. Stanford CA: Stanford University.
- van der Velde, F., & de Kamps, M. (2006). Neural blackboard architectures of combinatorial structures in cognition. *Behavioral and Brain Science*, 29, 37–70 and 88–108.
- Vigliocco, G., & Nicol, J. (1998). Separating hierarchical relations and word order in language production: Is proximity concord syntactic or linear? *Cognition*, 68, B13–B29.
- Vosse, T., & Kempen, G. (2000). Syntactic structure assembly in human parsing: a computational model based on competitive inhibition and a lexicalist grammar. *Cognition*, 75, 105–143.
- Vosse, T., & Kempen, G. (2008). Parsing verb-final clauses in German: Garden-path and ERP effects modeled by a parallel dynamic parser. In B. C. Love, K. McRae, & V. M. Sloutsky (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society (Washington DC)*. Austin, TX: Cognitive Science Society.
- Vosse, T., & Kempen, G. (2009a). The unification space implemented as a localist neural net: predictions and error-tolerance in a constraint-based parser. *Cognitive Neurodynamics*, 3, 331–346.
- Vosse, T., & Kempen, G. (2009b). In defense of competition during syntactic ambiguity resolution. *Journal of Psycholinguistic Research*, 38, 1–9.
- Woods, W. A. (1970). Transition network grammars for natural language analysis. *Communications of the ACM*, 13, 591–606.
- Ye, Z., Habets, B., Jansma, B. M., & Münte, T. F. (2011). Neural basis of linearization in speech production. *Journal of Cognitive Neuroscience*, 23, 3694–3702.
- Zylberberg, A., Slezak, D. F., Roelfsema, P. R., Dehaene, S., & Sigman, M. (2010). The brain's router: a cortical network model of serial processing in the primate brain. *PLoS Computational Biology*, 6, e1000765.