Assembly pointers for variable binding in networks of spiking neurons

Robert Legenstein[1], Christos H. Papadimitriou[2], Santosh Vempala[3], Wolfgang Maass[1]
[1] Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria, {legi,maass}@igi.tugraz.at.
[2] EECS, UC Berkeley, CA 94720, USA, christos@cs.berkeley.edu
[3] College of Computing, Georgia Tech, Atlanta, GA 30308, USA, vempala@gatech.edu

November 14, 2016

Abstract

We propose a model for binding of variables such as the thematic role of a word in a sentence or episode (e.g., agent or patient), to concrete fillers (e.g., a word or concept). Our model is based on recent experimental data about corresponding processes in the human brain. One source of information are electrode recordings from the human brain, which suggest that concepts are represented in the medial temporal lobe (MTL) through sparse sets of neurons (assemblies). Another source of information are fMRI recordings from the human brain, which suggest that subregions of the temporal cortex are dedicated to the representation of specific roles (e.g., subject or object) of concepts in a sentence or visually presented episode. We propose that quickly recruited assemblies of neurons in these subregions act as pointers to previously created assemblies that represent concepts. We provide a proof of principle that the resulting model for binding through assembly pointers can be implemented in networks of spiking neurons, and supports basic operations of brain computations, such as structured information retrieval and copying of information. We also show that salient features of fMRI data on neural activity during structured information retrieval can be reproduced by the proposed model.

1 Introduction

Numerous electrode recordings from the human brain (see [1] for a review) suggest that concepts are represented through assemblies of „concept cells“, i.e., sparse sets of neurons that fire (more or less) whenever the corresponding concept is activated. The data confirms earlier hypotheses and models, going back to [2] about the representation of tokens of cognitive computations through assemblies of neurons. More recent data [3] also provides also information about the way in which these assemblies are quickly modified in the human brain when we experience an association between two concepts. This data suggests that assemblies should not be seen as invariant entities, but as fluent coalitions of neurons (as proposed by [4]) whose response properties vary fast, even through a single experience. Furthermore the data of [3] suggests that this process on the level of neurons underlies the association of concepts or images, that for example supports recall of some image components (face or landmark) when an associated image component is presented. It is shown in [3] that some fraction of neurons in the assembly for one image component also starts to respond to the other image component. In other words, each of the two assemblies expands into the other assembly, so that the activation of one of them increases the activation probability for the other

assembly. This mechanism suggests that assemblies can act as pointers to other assemblies. They are uniquely qualified for this role, because in contrast to a single neuron, an assembly consisting of hundreds of thousands of neurons can trigger, through its activation, the firing of other neurons, and thereby also gate their plasticity. In addition, if assemblies are sufficiently large, a fair number of direct synaptic connections are likely to exist between the neurons of any two assemblies, even if the connection probability between any pair of neurons is low.

We propose that assemblies of neurons are also instrumental for creating a transient or longer lasting binding of a variable to a filler. For example, they could bind a variable that represents a thematic role (e.g., agent or patient in an episode) to a word or concept. Information about the neural representation of semantic roles is provided through recent fMRI data, where specific subregions in the temporal cortex were shown to respond to specific semantic (thematic) roles of individuals in an episode that was communicated through a sentence [5] or a movie [6].

Here we do not assume that semantic roles are represented by fixed assemblies of neurons. Such a fixed assembly would in general not have sufficient direct synaptic connectivity to the virtually unlimited repertoire of words or concepts, each represented through assemblies in other brain regions that could acquire this semantic role in an episode. To achieve such large potential connectivity, the size of this fixed assembly would have to be so large that its activation would not be consistent with generic sparse firing activity in each brain region. Rather, we propose that the specific subregions of the temporal cortex that were shown to be activated differentially in dependence of the specific semantic role of a concept serve as large pools of neurons (we will refer to them as neural spaces). In neural spaces, sparse assemblies can quickly be recruited from the subset of neurons that happen to have direct synaptic connections to a particular assembly in a region where content (concepts) are encoded, and thereby act as assembly pointers. We propose that this model can reconcile functional needs, such as being able to recall a concept from its recent thematic role, with data on the inherently sparse connectivity between brain areas [7]. One can also view this model as a direct extrapolation of data on the formation of associations between concepts from [3] to associations between thematic roles (i.e., variables) and concepts.

We propose that one well-known neurophysiological mechanism is essential for the control of this binding process: disinhibition. At least two different ways how brain areas or specific neural circuits can be selectively disinhibited have been proposed on the basis of experimental data [8]. One is neuromodulatory control (especially cholinergic), see [9]. Another one is disinhibition via the activation of VIP cells, i.e., of inhibitory neurons that primarily target other types of inhibitory neurons [10]. We propose that disinhibition plays a central role for neural computation and learning by controlling the creation and reactivation of assembly pointers.

Assembly pointers provide a model for a brain mechanism that replaces the "copy" operation that moves information in a digital computer. In contrast to a digital computer, the brain uses a "spatial code" for content, where the firing of a particular set of neurons (possibly widely distributed throughout the brain) indicates the recall of a content. Obviously, such a spatial code cannot be easily moved to another brain location. Hence, brain computations are based on a different paradigm where the copying of bit vectors is avoided altogether. Such alternative paradigms become important in computer science in efforts to create non-von-Neumann architectures that consume substantially less energy [11]: it is estimated that at least half of the energy of a current computer is consumed by shuffling of data between memory and processors. The assembly pointer concept may provide a biologically inspired basis for alternative computer architectures that are based on different relations between memory and processors.

2 Results

2.1 Variable binding through assembly pointers

A specific a word or a concept (referred to more abstractly as "content" in the following) is represented in our model by a specific assembly of neurons in a content space $\mathcal{C}$. This coding assumption is consistent with experimental data that arise through simultaneous recording from large sets of neurons, through multi-electrode arrays or Ca^{2+} imaging. The neural activity patterns that were found in this way can be characterized in first approximation as spontaneous and stimulus-evoked switching between the activations of different (but somewhat overlapping) subsets of neurons (see e.g. [4, 12, 13]), often referred to as assemblies of neurons.

The results and hypotheses of Frankland and Greene [5] provide the basis for our model for binding a variable v that represents a syntactic role (agent, verb, patient) to a concrete content (a word or a concept) in content space $\mathcal{C}$. We will refer to the particular region or set of neurons that is reserved for this variable as the neural space $\mathcal{N}_v$ for variable v. Each such neural space can be viewed as functioning like a register in a computer [5]. But in contrast to a computer, this "register" is not used for storing content in it. Rather, assemblies in this register $\mathcal{N}_v$ store "handles" or "pointers" to assemblies that store content information in the separate content space $\mathcal{C}$. The results of [5] indicate that disjoint subareas of temporal cortex represent different variables. We therefore represent different variables $v_1, \ldots, v_K$ in our model in separate neural spaces $\mathcal{N}_{v_1}, \ldots, \mathcal{N}_{v_K}$ for each of these variables. We refer to the union of the neural spaces for all variables as the variable space $\mathcal{V}$.

We do not assume specifically designed neural circuits that implement neural spaces and variable binding. Instead, we assume a rather generic network for each neural space and the content space, with lateral excitatory connections and lateral inhibition within the space. Further, we assume that neurons in the content space are sparsely connected to neurons in neural spaces for variables. We will show that the binding of variables to fillers emerges naturally in such a generic circuit model from plasticity processes.

In addition our model takes into account that neurons typically do not fire just because they receive sufficiently strong excitatory input. Experimental data suggest that neurons are typically prevented from firing by an "inhibitory lock", that balances or even dominates excitatory input [14]. Thus a generic pyramidal cell is likely to fire because two events take place: its inhibitory lock is temporarily lifted ("disinhibition") and its excitatory input is sufficiently strong. A special type of inhibitory neuron (VIP cells) has been identified as a likely candidate for triggering disinhibition, since VIP cells target primarily other types of inhibitory neurons (PV+ and SOM+ positive cells) that inhibit pyramidal cells, see e.g. [10]. Firing of VIP cells is apparently often caused by top-down inputs (they are especially frequent in layer 1, where top-down and lateral distal inputs arrive). Their activation is conjectured to enable neural firing and plasticity within specific patches of the brain through disinhibition, see e.g. [8, 15, 16, 17, 9]. One recent study also demonstrated that longterm plasticity in the human brain can be enhanced through disinhibition [18]. We propose that top-down disinhibitory control plays a central role for neural computation and learning in cortex by initiating for example the creation and reactivation of assembly pointers.

2.2 Creation of assembly pointers through STDP in disinhibited neural spaces

To test the proposed model of binding through assembly pointers, we performed computer simulations where stochastically spiking neurons were embedded in a corresponding network structure (see Fig. 1A and *Methods* for details). The network consisted of a content space $\mathcal{C}$ and a single neural space $\mathcal{N}_v$ for a variable v that each contained 1000 recurrently connected neurons (connection

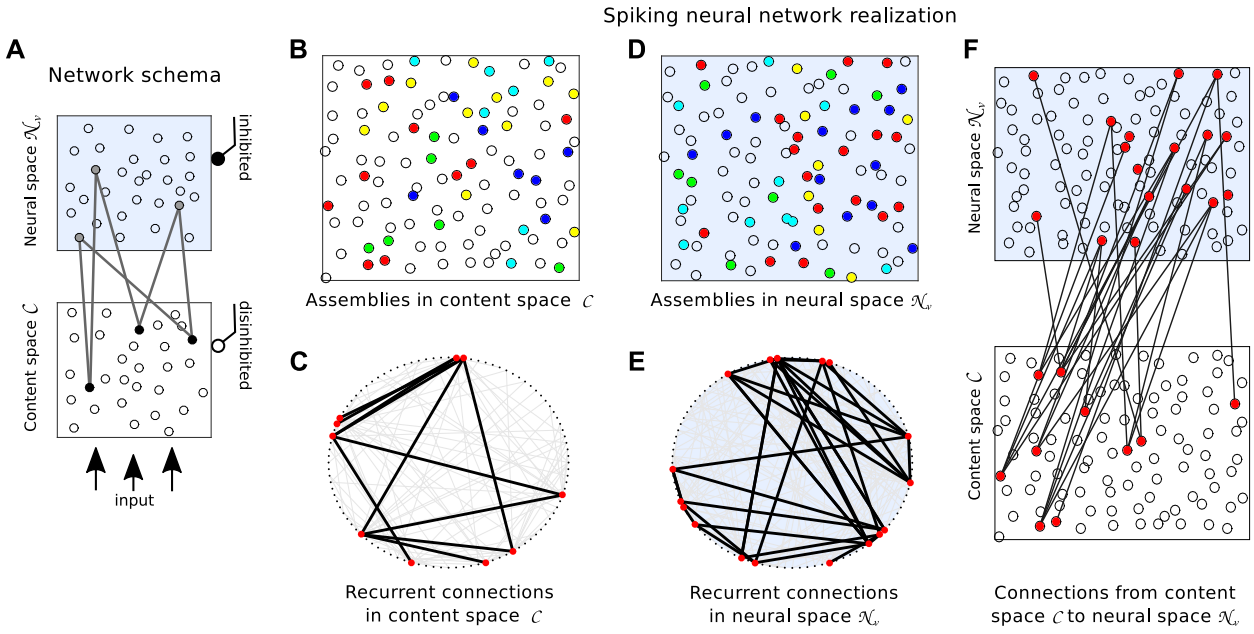

Figure 1: **Neural spaces and assembly pointers. A)** Network structure. Rectangles indicate the content space $\mathcal{C}$ (white) and the neural space $\mathcal{N}_v$ for variable v (light blue shading). Circles denote neurons (open: inactive; filled black: active; filled gray: potentially active but inhibited). Spaces $\mathcal{C}$ and $\mathcal{N}_v$ also include sparse recurrent connections which are not shown for clarity. Circle on top right of each space indicates disinhibition (filled black circle: inhibited; open circle: disinhibited). Gray lines indicate connections between spaces. Concepts are encoded in content space through neural assemblies. Initially, the neural space $\mathcal{N}_v$ is inhibited. Filled gray circles indicate neurons with connections from active neurons in $\mathcal{C}$. These neurons may become active when the neural space is disinhibited to constitute an assembly pointer **B)** Assembly code in content space after induction of assemblies through STDP in a spiking neural network (SNN) model. Black circles denote neurons in content space (100 randomly chosen out of 1000 and randomly placed in 2D space). Filling color indicates assembly identity. Open circles denote neurons which do not belong to any assembly. **C)** Assembly formation in content space of SNN. The 100 neurons shown in panel (B) are rearranged on a circle (black and red dots). Red dots denote neurons of the red assembly in (B) and thick black (light gray) lines strong (weak) connections between neurons (only connections to or from these assembly neurons are shown for clarity). Assemblies have strong inter-assembly connectivity and only weak connectivity to extra-assembly neurons. **D)** Assembly code in neural space $\mathcal{N}_v$ after a CREATE operation (as in panel B). **E)** Assembly formation in neural space (as in panel C). **F)** Connections from content space $\mathcal{C}$ to neural space $\mathcal{N}_v$ after a CREATE operation between the red assemblies from (B) and (D). Shown are all significantly strong connections (weights > 0.05 for a maximum weight of 0.5) from any shown neuron in content space $\mathcal{C}$ to any of the shown assembly neurons in neural space $\mathcal{N}_v$. Connections from neural space $\mathcal{N}_v$ to content space $\mathcal{C}$ were similar (not shown).

probability 0.1). In each of these spaces, lateral inhibition was implemented in a symbolic manner to ensure sparse activity. Disinhibition was modeled through a multiplicative effect of an inhibitory input on the membrane potential of neurons (see *Methods*). Reciprocal connections between $\mathcal{C}$ and $\mathcal{N}_v$ were introduced randomly with a connection probability of 0.1. Neurons in the content space received in addition connections from 200 input neurons. All synapses between excitatory neurons in the circuit were subject to spike-timing dependent plasticity (STDP).

First, we defined five simple rate patterns $P_1, \ldots, P_5$ that modeled the input to content space when a given concept or word (such as "truck" or "ball") is experienced. These patterns were repeatedly presented as input to the network (see *Methods*). Due to these pattern presentations, an assembly $\mathcal{C}(P_i)$ emerged in content space for each of the patterns P_i (assembly sizes between 81 and 86 neurons) that showed robust firing activity (> 50 Hz) whenever the corresponding pattern was presented as input, see Fig. 1B. STDP of recurrent connections led to a strengthening of these synapses within each assembly, while synapses between assemblies remained weak (see Fig. 1C and *Methods* for details).

During the induction of these assemblies in content space, the neural space $\mathcal{N}_v$ remained inhibited. We next simulated disinhibition of the neural space $\mathcal{N}_v$ while input to content space $\mathcal{C}$ excited an assembly there. Our model for variable binding assumes that such disinhibition enables the creation of an assembly pointer to the currently active assembly in the content space, whose neurons have synaptic connections to some neurons in this neural space, see Fig 1A. Such disinhibition of a neural space allows that some of neurons in it can fire, especially those that receive sufficiently strong excitatory input from a currently active assembly in the content space. Furthermore, in line with previously cited experimental reports we assume that this allowed firing of neurons in the neural space also enables plasticity of these neurons and synapses that are connected to it. In fact, STDP at the synapses that connected the content space $\mathcal{C}$ and the neural space $\mathcal{N}_v$ led to the stable emergence of an assembly $\mathcal{N}_v(P_i)$ in neural space within one second when some content P_i was represented in $\mathcal{C}$ during disinhibition of $\mathcal{N}_v$, see Fig. 1D, F. Further, plasticity at recurrent synapses in neural space $\mathcal{N}_v$ induced strengthening of recurrent connections within assemblies there, see Fig. 1E. Hence, disinhibition led to the rapid and stable creation of an assembly in the neural space $\mathcal{N}_v$, i.e., an assembly pointer. We denote such creation of an assembly pointer in a neural space $\mathcal{N}_v$ for a specific variable v to content P encoded in content space by CREATE(v, P).

Fast recruitment of assemblies in a neural space for some variable necessitates rapid forms of plasticity. We assumed that some (possibly initially transient) plasticity of neurons and/or synapses occurs instantaneously, even within seconds. Such assumption is usually not included in neural network models, but it is supported by a number of recent experimental data. In particular, [3] shows that neurons in higher areas of the human brain change their response to visual stimuli after few or even a single presentation of a new stimulus where two familiar images are composed into a single visual scene.

Our model for variable binding based on assembly pointers further assumes that strengthened synaptic connections between assemblies in neural space $\mathcal{N}_v$ for variable v and content space $\mathcal{C}$ enable the recall RECALL(v) of the variables' content, i.e., the activation of the assembly $\mathcal{C}(P)$ in content space that was active at the most recent CREATE(v, P) operation (e.g., representing the word "truck"). It has been shown that the excitability of pyramidal cells can be changed in a very fast but transient manner through fast depression of GABA-ergic synapses onto pyramidal cells [19]. This effect is potentially related to the match enhancement or match suppression effect that has been observed in neural recordings from monkeys, and is commonly used in neural network models for delayed match-to-sample (DMS) tasks, see e.g. [20]. Using such a mechanism, a RECALL(v) can be initiated by disinhibition of the neural space $\mathcal{N}_v$ while the content space does not receive any bottom up input, see Fig. 2A. The increased excitability of recently activated neurons in $\mathcal{N}_v$

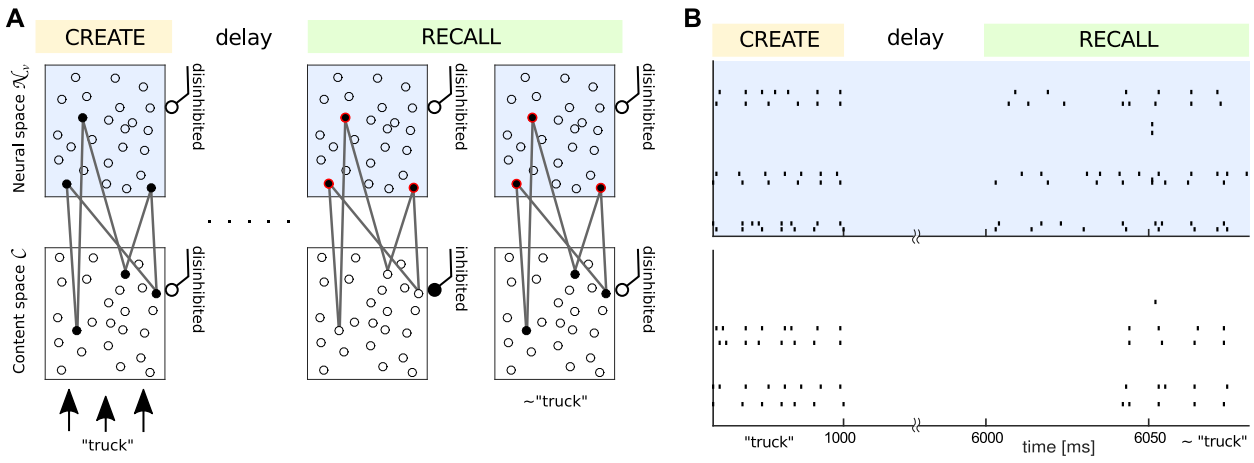

Figure 2: **Memory recall through assembly pointers. A)** Schematic of assembly pointer recall. After an assembly pointer was created (CREATE) for the word "truck", the excitability of assembly neurons in neural space $\mathcal{N}_v$ is enhanced (indicated by red color). When the neural space is disinhibited, these neurons are activated which in turn activate the "truck" assembly in content space $\mathcal{C}$ (RECALL). **B)** Spike rasters from neural space $\mathcal{N}_v$ (top) and content space $\mathcal{C}$ (bottom) in a simulated recall (every 20^{th} neuron shown for clarity). After a CREATE (left, up to 1 s), and a delay for 5 s, a RECALL is initiated by first disinhibiting the neural space $\mathcal{N}_v$ (at time $t = 6$ s) and then disinhibiting the content space $\mathcal{C}$ (40 ms later).

ensures that the most recently active assembly is activated which in turn activates the corresponding content through its (previously potentiated) feedback connections to content space $\mathcal{C}$.

Fig. 2B shows the spiking activity in our spiking neural network model for an example recall 5 seconds after the creation of the assembly pointer. A transient increase in neuron excitability has been included in the stochastic spiking neuron model through an adaptive neuron-specific bias current that increases slightly for each postsynaptic spike and decays with a time constant of 5 seconds (see *Methods* for details). We found that the content of the pointer can reliably be recalled. In general, recall performance was excellent. We considered recall after a 5 seconds delay for a model with 5 assemblies stored in content space and two neural spaces for variables. Testing separate recalls of the five patterns from each of these two neural spaces (i.e., 10 recalls), recall was perfect in 9 cases, i.e., assembly neurons in content space were active in the recall phase (firing rate greater or equal to 50 Hz) if and only if they were active during the creation of the assembly pointer. In one case, recall was close to perfect (here, only a single assembly neuron was not activated above 50 Hz during the recall).

2.3 Cognitive computations with assembly pointers

Apart from the creation of assembly pointers and recall of content, two further operations have been postulated to be essential for many higher cognitive functions [21]. The first is COPY(u, v) that copies the content of variable u to variable v. In our model, the copy operation creates an assembly pointer in neural space $\mathcal{N}_v$ for variable v to the content to which the assembly pointer in neural space $\mathcal{N}_u$ for variable u refers to. This operation can be implemented in our model simply by disinhibiting $\mathcal{N}_u$ in order to activate the corresponding content in $\mathcal{C}$ followed by a disinhibition of $\mathcal{N}_v$ in order to create an assembly pointer there, see Fig. 3A. We simulated this copy operation in our spiking neural network model with one content space and two neural spaces. The performance was tested

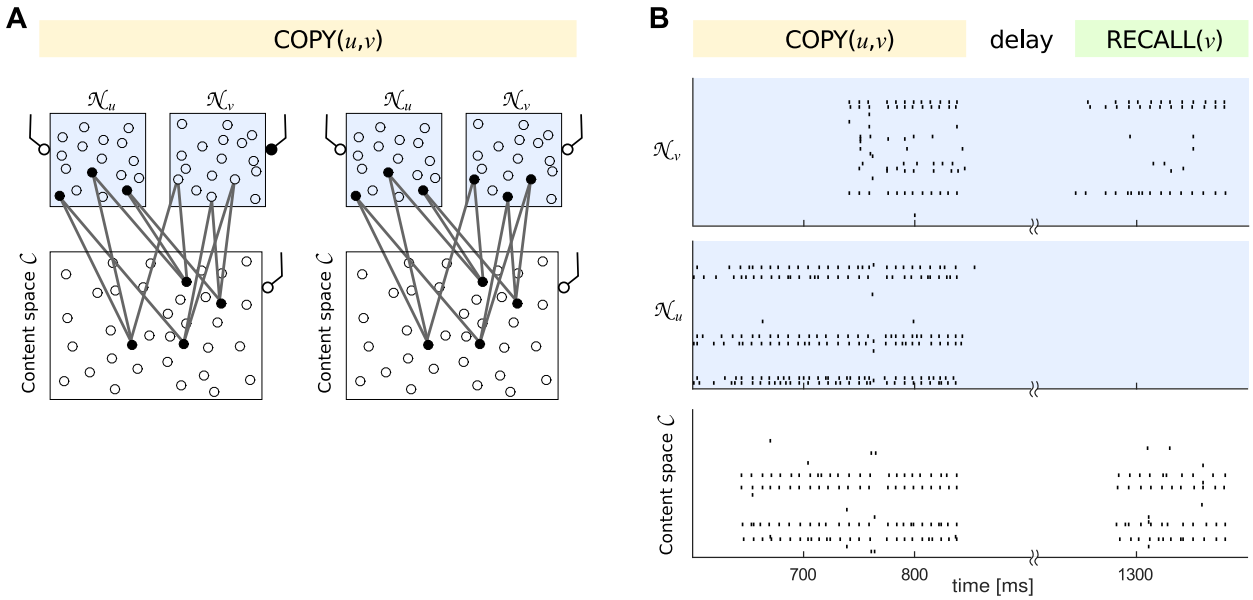

Figure 3: **Assembly pointer copy operation. A)** Schematic of assembly pointer copy. Disinhibition of neural space $\mathcal{N}_u$ recalls its content in content space $\mathcal{C}$ (left). A subsequent disinhibition of neural space $\mathcal{N}_v$ creates an assembly pointer for this content there (right). **B)** Spike rasters from neural spaces $\mathcal{N}_v$ (top), $\mathcal{N}_u$ (middle), and content space $\mathcal{C}$ (bottom) in a simulated copy operation from a variable u to a variable v ($600 - 840$ ms; every 20^{th} neuron shown for clarity). After a 400 ms delay, the content of variable v is tested by a recall at time 1240 ms. The assembly is correctly recalled in content space.

through a recall from the target assembly pointer 400 ms after the pointer content was copied, see Fig. 3B. We considered the same setup as described above where 5 assemblies were established in the content space. After copying each of these contents, recall was again close to perfect: 3 perfect recalls; 1 recall with one additional neuron activated above 50 Hz (while 81 assembly neurons were correctly activated); one recall with one assembly neuron activated below 50 Hz (while 86 assembly neurons were correctly activated).

A final fundamental operation considered in [21] is COMPARE(u, v) which compares whether the content of u equals the content of v. One possible implementation of this operation in our model is a readout neuron that receives depressing synaptic connections from the content space. Then, when the content for $\mathcal{N}_u$ and $\mathcal{N}_v$ is recalled in sequence, readout synapses will be depressed for the content of $\mathcal{N}_v$ if and only if the content of $\mathcal{N}_u$ equals the content of $\mathcal{N}_v$. Such a "change detecting" readout thus exhibits high activity if the contents of $\mathcal{N}_u$ and $\mathcal{N}_v$ are different, see Fig. 4A. Simulation results from our spiking neural network model are shown in Fig. 4B. They indicate that this simple mechanism is sufficient to compare assembly pointers perfectly simply by thresholding the activity of the readout after the recall from the second assembly pointer.

2.4 Reproducing experimental data on the binding of words to roles and structured information retrieval

Two experiments were performed in [5] that provided new insights in how variables may be encoded in neocortex. Sentences were shown to participants where individual words (like "truck" or "ball") can occur as the agent or as the patient. The authors then studied how cortex retrieves the information

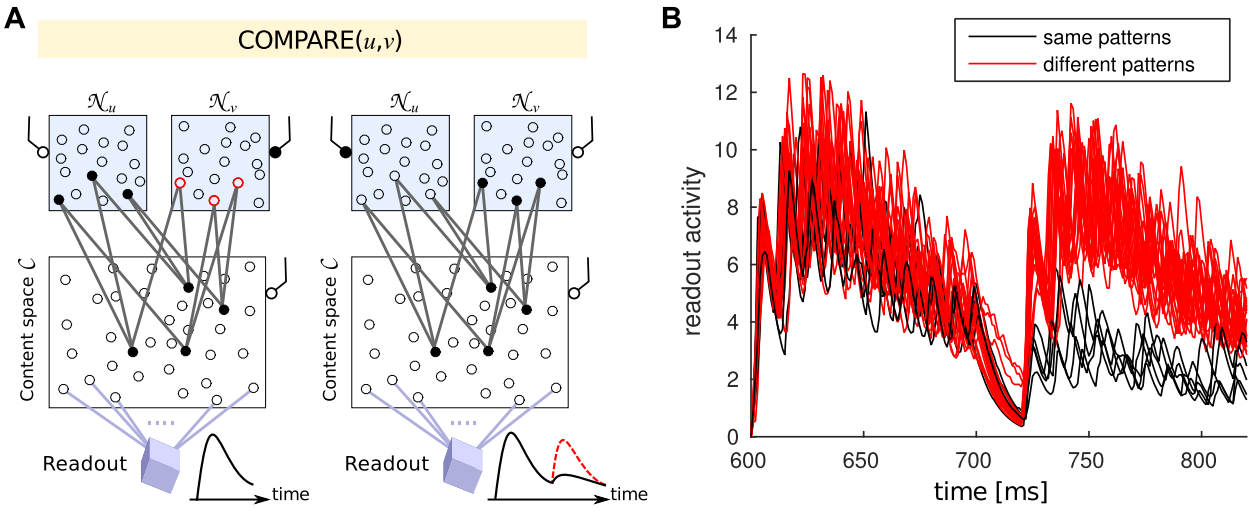

Figure 4: **Comparison with assembly pointers. A)** Schematic of comparison COMPARE(u, v). A readout neuron (violet) received depressing synaptic connections from all content space neurons. The comparison consists of a recall from $\mathcal{N}_u$ (left; red neurons in $\mathcal{N}_v$ indicate neurons with higher excitability) followed by a recall from $\mathcal{N}_v$ (right). During the first recall, readout weights become depressed and readout activity decreases (indicated by black trace right to the readout). Second recall shown for same pattern (right). Readout weights are still depressed and readout response is therefore weak (black trace at readout). If the content changes (i.e., $u \neq v$), readout weight from active neurons in $\mathcal{C}$ are not depressed, which leads to strong readout activity (red broken trace at readout). **B)** Comparisons in spiking neural network model. Each trace shows the activity of the readout neuron (arbitrary units) for one comparison operation between two assembly pointer contents (25 comparisons, one for each possibility how 5 contents can be bound to two neural spaces for variables u and v). At time 600 ms, the content of neural space $\mathcal{N}_u$ was recalled and the readout reacted in a similar manner to all contents. At time 720 ms, the content of neural space $\mathcal{N}_v$ was recalled. Due to depressed synaptic connections, the readout response was much weaker when the content of $\mathcal{N}_v$ matched the content of $\mathcal{N}_u$ (black traces) as compared to different contents in $\mathcal{N}_u$ and $\mathcal{N}_v$ (red traces).

contained in a sentence in a structured manner. In a first experiment, the authors aimed to identify cortical regions that encode the meaning of such sentences. Four example sentences with the words "truck" and "ball" are "The truck hit the ball" (S1), "The ball was hit by the truck" (S2), "The truck was hit by the ball" (S3), and "The ball hit the truck" (S4). Here, S1 and S2 (and S3 and S4 respectively) have the same meaning, which can be distinguished for example by answering the question "What was the role of the truck?". Indeed, the authors showed that a linear classifier is able to classify the meaning of such sentences from the fMRI signal of left mid-superior temporal cortex (lmSTC). Using our model for assembly pointers, we can model such situations by binding words either to an agent variable ("who did it") or to a patient variable ("to whom it was done"). Under the assumption that lmSTC hosts neural spaces (with assembly pointers) for the role of words, it is expected that the meaning of a sentence can be decoded from the activity there (Fig. 5; classifier 1), but not from the activity in content space where the identities are encoded independently of their role. We performed simulations where the words "truck" and "ball" (represented by some assemblies in content space) were sequentially bound (the temporal sequence was according to the position of the word in the sentence) either to neural space $\mathcal{N}_{\text{agent}}$ or $\mathcal{N}_{\text{patient}}$, depending on their role. Low-

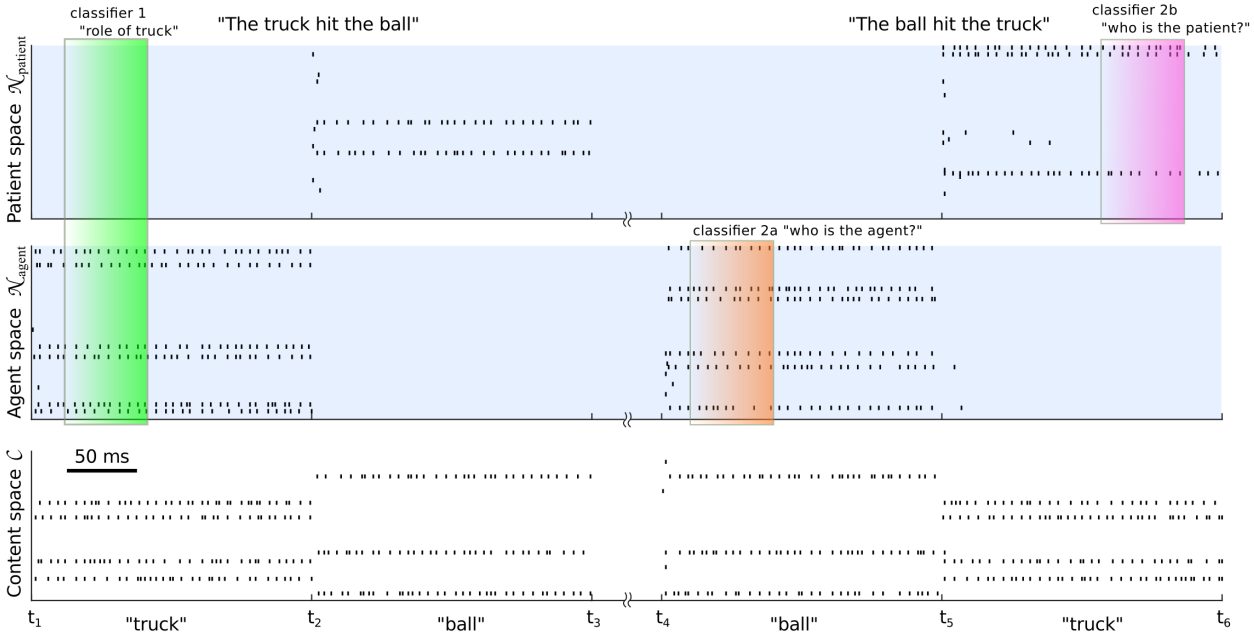

Figure 5: **Binding of words to roles, according to [5].** The content space (bottom, white) contained assemblies for five items (such as "truck" and "ball"). Words were presented (bottom) as they appear in the sentence (top), activating these assemblies. Words were bound to assembly pointers in two neural spaces (light blue) that represented the variables "agent" (middle) and "patient" (top). Spike rasters for the two example sentences "The truck hit the ball" (t_1 – t_3) and "The ball hit the truck" (t_4 – t_6) are shown. A linear classifier trained on the low-pass filtered spike response (indicated by green gradient) was able to classify sentences by their meaning (i.e., identify the role of the truck), modelling experiment 1 in [5]. Two additional classifiers were able to determine the current agent (orange gradient) and patient (magenta gradient) in sentences, modelling experiment 2 in [5].

pass filtered network activity was extracted for each of the sentences S1 to S4. We then trained a linear classifier to classify for each time point during the sentence presentation the meaning of the sentence based on a noisy version of filtered network activity. We found that even under severe noise conditions, the classifier was able to nearly perfectly classify test sentences (i.e., new simulations with new noisy activity on the same sentences; test classification error 0.5 %). On the other hand, a classifier based on activity of the content space performed only slightly better than random with a test classification error of 44 %.

A second experiment in [5] revealed that subregions of lmSTC also contain information about the current value of the variables for the agent and the patient. More specifically, the authors showed that one is able to predict from the fMRI signal of one subregion of lmSTC the identity of the agent and from the signal in another subregion the identity of the patient (generalizing over all identities of other roles and over different verbs). We expected that this would also be the case in the proposed model since the assemblies that are formed in the neural spaces $\mathcal{N}_{\text{agent}}$ and $\mathcal{N}_{\text{patient}}$ are typically specific to the bound content. We tested this hypothesis by training a multinomial logistic regression model to classify the content of the variable for each of the two neural spaces (agent and patient) at times when these spaces were disinhibited (Fig. 5; classifier 2a and classifier 2b). Here, we presented sentences to the model as above, but we considered all 40 possibilities of how 5 items

(words) $A_1, \ldots, A_5$ can be presented in a sentence (for example: A_1 is the agent and presented first, A_2 is the patient and presented as second item; we excluded sentences where the agent and patient is the same word). Low-pass filtered activity of a subset of neurons was sampled at every 10 ms to obtain the feature vectors to the classifiers (see *Methods*). Half of the sentences were used for testing where we made sure that the two items used in a given test sentence have never been shown in any combination in one of the sentences used for training. Consistent with the results in [5], the classifier achieved nearly optimal classification performance on test data (classification error < 2 % for both neural spaces). Note that such classification would fail if each neural space consisted of only a single assembly that is activated for all possible fillers [21], since in this case no information about the identity of the role is available in the neural space for the variable.

3 Discussion

It has often been emphasized (see e.g. [22, 23]) that there is a need to understand brain mechanisms for variable binding. We propose in this article a model for variable binding through "assembly pointers". Our model is consistent with recent findings on cortical assemblies and the encoding of sentence meaning in cortex [5]. This model is not based on the construction of specific neural circuitry to perform this binding. Instead, it is based on generic sparsely and randomly connected neural spaces that organize their computation based on fast plasticity mechanisms. The model provides a direct link between information processing on the computational level of symbols and sentences and processes on the implementation level of neurons and synapses. The resulting model for brain computation supports top down structuring of incoming information, thereby laying the foundation of goal oriented „willful" information processing rather than just input-driven processing. The proposed synaptic plasticity that links assemblies in different neural spaces can be transient, but could also become more permanent if its relevance is underlined through repetition and consolidation. This would mean that some neurons in the neural space for a variable are no longer available to form new pointer assemblies, but this is no problem if the neural space for each variable is sufficiently large.

Several models for variable binding had been proposed in the literature. In general, these models fall into one of the general classes of pointer-based binding, binding by synchrony, or convolutional binding. Pointer-based models (e.g., [21, 24]) assume that pointers are implemented by single neurons or populations of neurons which are activated as a whole group. In contrast, our model is based on the assumption that distributed assemblies of neurons are the fundamental tokens for encoding symbols and content in the brain, and also for pointers. We propose that these assembly pointers can be created on the fly in some neural spaces for variables and occupy only a sparse subset of neurons in these spaces. It has been shown in [5] that the filler of a thematic role (e.g. the actor) can be predicted from the fMRI signal of a subregion in temporal cortex when a person reads a sentence. As shown above, this finding is consistent with assembly pointers. It is however inconsistent with models where a variable engages a population of neurons that is independent of the bound content, such as traditional pointer-based models. In comparison to traditional pointer models, the assembly pointer model could also give rise to a number of functional advantages. In a neural space $\mathcal{N}_v$ for a variable v, several instantiations of the variable can coexist at the same time, since they can be represented there by increased excitabilities of different assemblies. These contents could be recalled as different possibilities in a structured recall and combined in content space $\mathcal{C}$ with the content of other variables to in order to answer more complex questions.

Some data shows that the relation between spiking activity and the phases of underlying oscillatory population activity may play a role in hippocampus and for working memory [25], indicating a possible role of synchrony in the binding process. Still, the reliability and capacity of binding

by synchrony is currently unclear. We note that, while our model is not based on precise synchronization of spikes in different neural spaces, the synaptic coupling between these spaces together with lateral inhibition leads to some synchronized oscillations of interacting neural spaces in our simulations. This is consistent with recent experimental data which suggest that common rhythms in two brain areas support the flow of excitation between these two areas, and also the potentiation of synapses between activated neurons in both areas [26].

Convolutional binding (see e.g., [27]) uses mathematical operations on high-dimensional vectors for variable binding. It had been used in the semantic pointer architecture of Eliasmith [28] where spiking neural networks were constructed to perform these rather complex operations. Similarly, the neural blackboard architecture (NBA, see e.g. [29]) relies on a number of neural circuits that were constructed for example to gate activity or to memorize associations. In contrast to these models, the assembly pointer model focuses on the emergence of binding operations, using assumptions on the fundamental level of assembly coding, network connectivity statistics, and plasticity processes.

The validity of the assembly pointer model for variable binding could be tested experimentally, since it predicts quite unique network dynamics during mental operations. First, binding of a variable to a concept employs disinhibition of a neural space related to that variable. This could be implemented by the activation of inhibitory VIP cells which primarily target inhibitory neurons, or by neuromodulatory input. Similar disinhibition mechanisms would be observed during a recall of the filler for that variable. Another prediction of the model is that a significant modification of the assembly that encodes a concept will also modify the pointer assembly to it that emerges in a neural space for some variable. Further, our model suggests that inactivation of a pointer assembly to some content A in neural space $\mathcal{N}_v$ would not abolish the capability to create a binding of the associated variable v to this content A: If the trial that usually creates this binding is repeated, a new pointer assembly in the neural space for v can emerge. Finally, the model predicts that a mental task that requires to copy (or compare) the filler of one variable u to another variable v causes sequential activation (disinhibition) of the neural spaces $\mathcal{N}_u$ and $\mathcal{N}_v$ for these variables.

We have presented a model for variable binding based on assembly pointers. The model is consistent with recent experimental data on assembly representations [1] in cortex and the representation of thematic roles in lmSTC [5]. Assembly pointers can reconcile functional needs, such as the recall of concepts, with data on the inherently sparse connectivity between brain areas [7] and sparse network activity.

4 Methods

General network architecture: The network consists of one content space $\mathcal{C}$ and (one or several) neural spaces $\mathcal{N}_{v_1}, \mathcal{N}_{v_2}, \ldots$ for variables $v_1, v_2, \ldots$. Each space consists of a number of spiking neurons ($N = 1000$ for all spaces in our simulations). Within each space, neurons are connected by sparse recurrent connections with $p = 0.1$ (i.e., for each pair of neurons, a connection between them is established with probability p; no self-connections are allowed). Neurons in $\mathcal{N}_{v_i}$ receive sparse excitatory connections ($p = 0.1$) from neurons in $\mathcal{C}$ and vice versa. These connections are symmetric, but the weights are not necessarily symmetric. Neurons in $\mathcal{C}$ additionally receive input from an input population ($N_{\text{in}} = 200$ in our simulations). Network dynamics was simulated in discrete time with a time step of $\Delta t = 1$ ms.

Neuron model: We used a single neuron model for all neurons in our simulations. In this model, the probability of a spike of neuron i at time t is given by the value of an activation variable $u_i(t)$ that models in an abstract way the membrane potential of the neuron. Once a neuron has spiked, its activation variable is reset to 0 and it enters a refractory period with a duration that is chosen

uniformly in $\{1, \ldots, 6\}$ ms. During the refractory period, the activation variable is clamped to 0. When the neuron is not refractory, the activation variable evolves according to

$$u_i(t) = \left(1 - \frac{\Delta t}{\tau_m}\right) u_i(t - \Delta t) + \left(\frac{\Delta t}{\tau_m}\right) G(t) \left(\exp\left(I_i(t) - I_{\text{Inh}}(t - \Delta t)\right) - 1\right), \tag{1}$$

where $\tau_m = 10$ ms is the membrane time constant, $G(t)$ denotes the inhibition status of the neural space (0 for fully inhibited, 1 for fully disinhibited), $I_i(t)$ is the input current to the neuron and $I_{\text{Inh}}(t)$ is the inhibitory current to the neurons in the neural space. In addition, the activation variable $u_i(t)$ is clipped at 0 and 1 at each time step. The inhibitory current is given by

$$I_{\text{Inh}}(t) = \left(1 - \frac{\Delta t}{\tau_{\text{Inh}}}\right) I_{\text{Inh}}(t - \Delta t) + \left(\frac{\Delta t}{\tau_{\text{Inh}}}\right) G(t) \left[\sum_k u_k(t) - \Theta_{\text{Inh}}\right]_{-2}^{4}, \tag{2}$$

where $\tau_{\text{Inh}} = 25$ ms and the sum runs over all neurons in the neural space. Θ_{Inh} determines the average activity in the neural space. It was set to $\Theta_{\text{Inh}} = 50/7$ in all spaces. The notation $[\cdot]_{-2}^{4}$ denotes that the argument (change of inhibition) was clipped at -2 and 4 to avoid instabilities. The input current I_i to neuron i is given by

$$I_i(t) = \sum_{j \in \mathcal{S}_i^{\text{FF}}} w_{ij} z_j(t) + \sum_{k \in \mathcal{S}_i^{\text{FB}}} w_{ik} z_k(t - \Delta t) + \sum_{l \in \mathcal{S}_i^{\text{REC}}} w_{il} z_l(t - \Delta t) + b_i(t), \tag{3}$$

where $z_j(t) \in \{0, 1\}$ denotes the spike output of neuron j at time t, w_{ij} denotes the synaptic weight from neuron j to neuron i, and $b_i(t)$ denotes the value of the adaptive excitability of neuron i at time t. $\mathcal{S}_i^{\text{FF}}$, $\mathcal{S}_i^{\text{FB}}$, and $\mathcal{S}_i^{\text{REC}}$ denote the sets of neurons that connect to neuron i in a feed-forward way (from inputs or from $\mathcal{C}$ to $\mathcal{N}_v$), as feedback (from some $\mathcal{N}_v$ to $\mathcal{C}$), or recurrently (within the neural space) respectively. To avoid instabilities, the input current was clipped at a value of 8. The excitability $b_i(t) \in [0, 1]$ decays exponentially with a factor $\tau_b = 5$ s and is increased with every spike of neuron i by $0.05(1 - b_i(t))$.

Plasticity equations: A simple model for STDP was used for all excitatory connections. In this model, the weight change at synapse ij for a post-synaptic spike at time t is given by

$$\Delta w_{ij}(t) = \eta \sum_{k : t_j^{(k)} < t} \left(\exp\left(\frac{t - t_j^{(k)}}{\tau_+}\right) - A_-\right), \tag{4}$$

where η is a learning rate, the sum runs over recent presynaptic spikes, $\tau_+ = 20$ ms is the time constant of the positive STDP window, and $A_- = 0.35$ is a negative offset that determines the amount of depression ($A_- = 0.35$ for feed-forward and recurrent connections and 0.1 for feedback connections). This rule is similar to [30], but without a weight dependency. The learning rate in $\mathcal{C}$ was set to 10^{-3} (feed-forward weights), $2.5 \cdot 10^{-4}$ (recurrent weights), and $5 \cdot 10^{-3}$ (feedback weights). In $\mathcal{N}_v$, the learning rate was $5 \cdot 10^{-3}$ for all weights. All weights were constrained to be non-negative. Feed-forward weights in $\mathcal{C}$ were clipped at 0.8, those in V_i at 0.5. Recurrent weights were clipped at 0.25 in $\mathcal{C}$ and at 0.2 in $\mathcal{N}_v$. Feedback weights were clipped at 0.25.

Plasticity was enabled in excitatory recurrent connections of the content space and input connections to content space during the initial formation of content assemblies (see below). Excitatory recurrent connections in the neural spaces for variables, input connections from content space to neural spaces for variables, and feedback connections from neural spaces for variables to the content space were plastic during CREATE and reload epochs (see below for the definition of reload epochs).

Change-detector readout: The output $z^{\mathrm{CD}}(t)$ of the change detector neuron at time t is given by $z^{\mathrm{CD}}(t) = \alpha^{\mathrm{CD}} z^{\mathrm{CD}}(t-\Delta t) + (1-\alpha^{\mathrm{CD}}) \sum_i w_i^{\mathrm{CD}}(t) z_i(t)$, where the sum runs over all neurons in the content space. $\alpha^{\mathrm{CD}} = 0.9$ is a filtering constant (time constant of 10 ms). The weights $w_i^{\mathrm{CD}}(t)$ are short-term depressing: $w_i^{\mathrm{CD}}(t) = w_i^{\mathrm{CD}}(t-\Delta t) + \eta_{\mathrm{CD}}(1 - w_i^{\mathrm{CD}}(t-\Delta t) - 10 z_i(t))$, with $\eta_{\mathrm{CD}} = 0.01$.

Initial formation of content assemblies: First, the content space learned to represent 5 very simple patterns presented at 200 input neurons. Each pattern consisted of 25 active input neurons that produced Poisson spike trains at 100 Hz while other neurons were silent, and each input neuron was active in at most one pattern. This initial learning phase consisted of 200 pattern presentations, where in each presentation, one pattern was chosen randomly from the set of five patterns. It was presented for 200 ms, followed by a 200 ms blank period where all input neurons were firing at 12.5 Hz, another pattern presentation, and so on.

For Fig. 1B, we presented each input pattern P_i separately to the network for 600 ms after the initial learning period. We then measured the firing rate of each neuron from $t = 100$ ms to $t = 600$ ms. A neuron was classified to belong to assembly A_i (for input P_i), if its firing rate during that time was above 50 Hz. In Fig. 1C, all potential connections from or to neurons in the depicted assembly (i.e., those neurons indicated by red dots) were drawn as gray lines. Thick black lines indicate weights that are larger than 0.05 (where the maximum weight was 0.25).

After these assemblies have been formed, plasticity of recurrent synaptic connections and those from input neurons was disabled.

Creation of assembly pointers (CREATE-operation): We next added two neural spaces $\mathcal{N}_u$ and $\mathcal{N}_v$ for two variables u, v with $N = 1000$ neurons per space. To induce stable assembly configurations in the neural spaces, we presented each input pattern for 1 s while the content space and one neural space were disinhibited. Each pattern was presented twice, with either $\mathcal{N}_u$ or $\mathcal{N}_v$ disinhibited. Neural assemblies in neural spaces for variables for Fig. 1D were defined as in content space, see above. Recurrent connection weights in Fig. 1E were drawn as in Fig. 1C.

Recall of content space assemblies (RECALL-operation): We next tested whether RECALL operations can reliably be performed by this circuit. For this test, a pattern was first presented to the network for 200 ms with one of the neural spaces $\mathcal{N}_u$ or $\mathcal{N}_v$ disinhibited. This corresponds to a brief (i.e. 200 ms) CREATE operation. Note that because assemblies in these spaces were already created previously (see above), previously potentiated synapses were still strong. Hence, the shorter presentation period was sufficient to activate the assembly in the neural space for the variable. We refer to such a brief CREATE in the following as a loading operation. After this loading epoch, a delay epoch of 5 s followed (no input presented). In order to make sure that no memory was kept in the recurrent activity, all spaces were inhibited in this period. After the delay, a recall-epoch followed. The recall epoch lasted for 140 ms during which the neural space $\mathcal{N}_u$ (or $\mathcal{N}_v$) was disinhibited. During the first 40 ms of this epoch, the content space stayed inhibited.

Copying of assembly pointers (COPY-operation): After a content was loaded into $\mathcal{N}_u$ and a brief delay epoch (400 ms), a RECALL operation was performed from $\mathcal{N}_u$ (140 ms as above). Then, in addition $\mathcal{N}_v$ was disinhibited for 100 ms. To test performance, a RECALL was initiated from $\mathcal{N}_v$ 400 ms later.

Comparison of assembly pointers (COMPARE-operation): Finally, we tested COMPARE operations in the circuit. We added a single linear neuron with depressing synapses as a change-detector readout from content space. The neuron received synapses from all neurons in content

space with uniform weights. Synaptic connections were subject to simple short-term depression (see above). To test the COMPARE operation, we first loaded a content A into neural space $\mathcal{N}_u$ and after a short delay a content B into neural space $\mathcal{N}_v$. After another delay of 200 ms, the pointers were compared. The COMPARE operation was implemented by two consecutive RECALL operations from $\mathcal{N}_u$ and $\mathcal{N}_v$. We performed comparisons between all 25 pairs of the 5 stored patterns.

Details to the binding of words to roles: These experiments modeled the findings in [5] about the binding of agents to roles in temporal cortex. We used again the network described above with one content space $\mathcal{C}$ and two neural spaces to which we refer in the following as $\mathcal{N}_{\text{agent}}$ and $\mathcal{N}_{\text{patient}}$. Input patterns to the network were interpreted as words in sentences. We used the network described above with 5 assemblies in $\mathcal{C}$ that represented 5 items (words) $A_1, \ldots, A_5$ and 5 assembly pointers in each neural space (created as described above). We defined that A_1 represents "truck" and A_2 represents "ball". We considered the four sentences "The truck hit the ball" (S1), "The ball was hit by the truck" (S2), "The truck was hit by the ball" (S3), and "The ball hit the truck" (S4). The processing of a sentence was modeled as follows. The words "truck" and "ball" were presented to the network (i.e., the corresponding input patterns) in the order of appearance in the sentence, each for 200 ms without a pause in between. During the presentation of a word, the activated assembly in $\mathcal{C}$ was bound to $\mathcal{N}_{\text{agent}}$ if it served the role of the agent and to $\mathcal{N}_{\text{patient}}$ if its role was the patient. For example, for the sentence "The truck hit the ball", first "truck" was presented and bound to $\mathcal{N}_{\text{agent}}$ (the "agent" variable), then "ball" was presented and bound to $\mathcal{N}_{\text{patient}}$ (the "patient" variable). The sequence of sentences S1 to S4 was presented twice to the network. The classifier described in the following was trained on the first sequence and tested on the second sequence.

Spiking activity was recorded in all neural spaces. Spiking activity was low-pass filtered with a filter time constant of 20 ms. Hence, for each neuron i we obtained its filtered activity r_i by

$$r_i(t) = \int_0^{100\text{ ms}} e^{-\frac{s}{20\text{ ms}}} S_i(t-s) \quad ds, \tag{5}$$

where S_i represents the spike train of neuron i in the form of a sum of Dirac delta pulses at spike times. Note that time was discretized with $\Delta t = 1$ ms. Independent zero-mean Gaussian noise of unit variance was added to each filtered activity at each time point. We denote by $\mathbf{r}_{\text{agent}}(t)$, $\mathbf{r}_{\text{patient}}(t)$, $\mathbf{r}_{\text{V}}(t)$, and $\mathbf{r}_{\text{C}}(t)$ the vector of filtered activities at time t from all neurons in neural space $\mathcal{N}_{\text{agent}}$, in neural space $\mathcal{N}_{\text{patient}}$, in both neural spaces, and in content space respectively.

The task for the first classifier ("role of truck") was to classify at each time point t the meaning of the current sentence (this is equivalent to determining the role of the truck). Hence, the sentences S1 and S2 constituted class $\mathcal{C}_0$ and sentences S3 and S4 the class $\mathcal{C}_1$. The classification was based on the current filtered network activity $\mathbf{r}_{\text{V}}(t)$ from the neural spaces (where neurons that were never active during any sentence were discarded). We used a simple linear model that was trained by linear regression with targets -2 for class $\mathcal{C}_0$ and 2 for class $\mathcal{C}_1$. An input vector was classified as belonging to class $\mathcal{C}_1$ if the output of the linear model was larger or equal than zero. For comparison, a classifier was also trained in the same manner on filtered network activity $\mathbf{r}_{\text{C}}(t)$ from content space.

To model the second experiment in [5], we considered sentences that were formed by tuples from the set of all five items $A_1, \ldots, A_5$, see *Results*. Then, the task for a second classifier ("who is the agent") was to classify from subsampled filtered network activity $\hat{\mathbf{r}}_{\text{agent}}(t)$ the identity of the current agent during those times when $\mathcal{N}_{\text{agent}}$ was disinhibited. Here, $\hat{\mathbf{r}}_{\text{agent}}(t)$ consisted of a subsample of the activities in $\mathbf{r}_{\text{agent}}(t)$ (these contained Gaussian noise as described above). To arrive at this subsample, we first discarded neurons that had an average activity below 10 Hz in the reload phases averaged over all sentences (mean activities during these phases were between 0 and 18 Hz with a strong peak close to 0 and a second mode at around 15 Hz). From the remaining neurons, we

selected every 4^{th} neuron to contribute to $\hat{\mathbf{r}}_{\text{agent}}(t)$. This procedure reduced the dimensionality of the feature vectors to 142 which significantly speeded up the fitting process of the model. Samples of the filtered activity were taken every 10 ms to arrive at the data set. The data set was divided into a training set and a test set as described in *Results*. We then fitted a nominal multinomial logistic regression model to the training set using the function mnrfit in MATLAB 8.5 and tested the model on the test set. Finally, the task for a third classifier ("who is the patient") was to classify from subsampled filtered network activity $\hat{\mathbf{r}}_{\text{patient}}(t)$ the identity of the current patient during those times when $\mathcal{N}_{\text{patient}}$ was disinhibited. The procedure was analogous to the procedure for the second classifier.

Acknowledgments

Written under partial support by the European Union project #604102 (Human Brain Project). We thank Adam Marblestone for helpful comments.

References

[1] R. Q. Quiroga. Neuronal codes for visual perception and memory. *Neuropsychologia*, 83:227–241, 2016.

[2] D. O. Hebb. *The Organization of Behavior*. Wiley, New York, 1949.

[3] M. J. Ison, R. Q. Quiroga, and I. Fried. Rapid encoding of new memories by individual neurons in the human brain. *Neuron*, 87(1):220–230, 2015.

[4] G. Buzsaki. Neural syntax: cell assemblies, synapsembles, and readers. *Neuron*, 68(3):362–385, 2010.

[5] S. M. Frankland and J. D. Greene. An architecture for encoding sentence meaning in left mid-superior temporal cortex. *Proceedings of the National Academy of Sciences*, 112(37):11732–11737, 2015.

[6] J. Wang, V. L. Cherkassky, Y. Yang, K. K., Chang, R. Vargas, N. Diana, and M.A. Just. Identifying thematic roles from neural representations measured by functional magnetic resonance imaging. *Cognitive Neuropsychology*, 33(3-4):257–264, 2016.

[7] X. J. Wang and H. Kennedy. Brain structure and dynamics across scales: in search of rules. *Current opinion in neurobiology*, 37:92–98, 2016.

[8] J. J. Letzkus, S. B. E. Wolff, and A. Lüthi. Disinhibition, a circuit mechanism for associative learning and memory. *Neuron*, 88:264–276, 2015.

[9] R. C. Froemke and C. E. Schreiner. Synaptic plasticity as a cortical coding scheme. *Current Opinion in Neurobiology*, 35:185–199, 2015.

[10] K. D. Harris and G. M. G. Shepherd. The neocortical circuit: themes and variations. *Nature Neuroscience*, 18(2):170–181, 2015.

[11] J. E. Kelly III and S. Hamm. *Smart Machines: IBM's Watson and the Era of Cognitive Computing*. Columbia University Press, 2013.

[12] B. Bathellier, L. Ushakova, and S. Rumpel. Discrete neocortical dynamics predict behavioral categorization of sounds. *Neuron*, 76(2):435–449, 2012.

[13] A. Luczak and J. N. MacLean. Default activity patterns at the neocortical microcircuit level. *Frontiers in Integrative Neuroscience*, 6(30):doi: 10.3389/fnint.2012.00030, 2012.

[14] B. Haider, M. Häusser, and M. Carandini. Inhibition dominates sensory responses in the awake cortex. *Nature*, 493(7430):97–100, 2013.

[15] P. Caroni. Inhibitory microcircuit modules in hippocampal learning. *Current Opinion in Neurobiology*, 35:66–73, 2015.

[16] C. K. Pfeffer. Inhibitory neurons: VIP cells hit the brake on inhibition. *Current Biology*, 24(1):R18–R20, 2014.

[17] Y. Fu, M. Kaneko, Y. Tang, a. Alvarez-Buylla, and M. P. Stryker. A cortical disinhibitory circuit for enhancing adult plasticity. *Elife*, 4:e05558, 2015.

[18] R. F. Cash, T. Murakami, R. Chen, G. W. Thickbroom, and U. Ziemann. Augmenting plasticity induction in human motor cortex by disinhibition stimulation. *Cerebral Cortex*, 26(1):58–69, 2016.

[19] D. M. Kullmann, A. W. Moreau, Y. Bakiri, and E. Nicholson. Plasticity of inhibition. *Neuron*, 75(6):951–962, 2012.

[20] E. M. Tartaglia, N. Brunel, and G. Mongillo. Modulation of network excitability by persistent activity: how working memory affects the response to incoming stimuli. *PLoS Comput Biol*, 11(2):e1004059, 2015.

[21] A. D. Zylberberg, L. Paz, P. R. Roelfsema, S. Dehaene, and M. Sigman. A neuronal device for the control of multi-step computations. *Papers in Physics*, 5:050006, 2013.

[22] G. F. Marcus. The algebraic mind: Integrating connectionism and cognitive science. *MIT Press*, 2003.

[23] G. F. Marcus, A. Marblestone, and T. Dean. The atoms of neural computation - does the brain depend on a set of elementary, reusable computations? *Science*, 346(6209):551–552, 2014.

[24] T. Kriete, D. C. Noelle, J. D. Cohen, and R. C. O'Reilly. Indirection and symbol-like processing in the prefrontal cortex and basal ganglia. *Proceedings of the National Academy of Sciences*, 110(41):16390–16395, 2013.

[25] M. Siegel, M. R. Warden, and E. K. Miller. Phase-dependent neuronal coding of objects in short-term memory. *Proceedings of the National Academy of Sciences*, 106(50):21341–21346, 2009.

[26] A. D. Friederici and W. Singer. Grounding language processing on basic neurophysiological principles. *Trends in Cognitive Sciences*, 19(6):329–338, 2015.

[27] T. A. Plate. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6(3):623–641, 1995.

[28] C. Eliasmith, T. C. Stewart, X. Choo, T. Bekolay, T. DeWolf, Y. Tang, and D. Rasmussen. A large-scale model of the functioning brain. *science*, 338(6111):1202–1205, 2012.

[29] F. Van der Velde and M. De Kamps. Neural blackboard architectures of combinatorial structures in cognition. *Behavioral and Brain Sciences*, 29(01):37–70, 2006.

[30] B. Nessler, M. Pfeiffer, and W. Maass. STDP enables spiking neurons to detect hidden causes of their inputs. *Proceedings of NIPS 2009: Advances in Neural Information Processing Systems*, 22:1357–1365, 2010.