

On the Natural Hierarchical Composition of Cliques in Cell Assemblies

Thomas Wennekers

© Springer Science+Business Media, LLC 2009

Abstract Hebbian cell assemblies can be formalised as sets of tightly connected cells in auto- and hetero-associative memories. Direct evidence for such “cliques” has recently been obtained in multiple-unit recordings from rat hippocampal neurons. These experiments suggest a hierarchical organisation where cliques are embedded in each other such that larger cliques represent less specific stimulus conditions. We here suggest an interpretation stating that the firing patterns may not just reflect nested categories but a lattice of concepts about stimulus–response mappings in the sense of formal concept analysis, an applied branch of set theory. We present an implementation of formal concept lattices in bidirectional associative memories that in contrast to previous work satisfies Dale’s principle and uses balanced excitation and inhibition. Inhibitory cells have fixed, non-plastic synapses even if the model learns new concepts. As an extreme case a single global inhibitory cell is enough that controls the total level of activation. The excitatory cells can further learn incrementally using a Hebbian coincidence learning rule. Implications of the model for retrieval in auto-associative memories are further outlined. Overall the model is well suited for representing hierarchical compositional relationships between entities in the form of correlated patterns in technical cognitive systems and potentially the brain.

Keywords Cell assemblies · Formal concept analysis · Associative memory · Clique codes · Bio-inspired computing

T. Wennekers (✉)
Centre for Theoretical and Computational Neuroscience,
School of Psychology, University of Plymouth, Portland Square
Building, Room A218, PL4 8AA Plymouth, Devon, UK
e-mail: Thomas.Wennekers@plymouth.ac.uk

Introduction

Cognitive Computation aims at developing the next generation computing hard- and software. On one hand it relies on the most advanced existing computer technologies for large-scale simulations of real and artificial neural systems—these, however, it wants to improve and overcome by brain-inspired neural future hardware [25, 61, 50]. On the other hand Cognitive Computation aims at a breakthrough in software technology as well, by developing artificial cognitive systems at a level of sophistication eventually adequate for communication with humans [29, 30, 57, 65]. This bilateral nature of Cognitive Computation, the combination of physics and function or hardware and software is nicely paralleled by one of the predominating theories of cortical computation, the theory of Hebbian cell assemblies which is our main focus of research. Originally laid out by Donald Hebb in the middle of the last century as the fundament of the then new discipline neuropsychology it still provides a strong stream in modern computational neuroscience today [22].

Cell assemblies are seen by their believers as the basic building blocks of cortical representations and meant to cover representations of any rememberable thing, ranging from concrete objects to abstract entities of thought [3, 6, 22, 38, 41, 55, 59, 60]. According to Hebb, neurons of one assembly are connected more strongly mutually than with other neurons whereby they can lend support to each other’s activation. If a sufficient part of an assembly is excited, say by a stimulus, the whole assembly can “ignite” and the remaining neurons be associatively completed. This basic mechanism would explain associative processes but also short-term memory in form of reverberating activity. Moreover, according to the slogan of Hebbianists “what fires together wires together” assemblies can form by synaptic

plasticity such that coactivation of neurons increases the connectivity within an assembly (long-term potentiation) but activity in active neurons decouples them synaptically from non-active neurons (long-term depression). Because at least a core of neurons should get invariantly activated if the same object is seen repeatedly (though perhaps with some variation), Hebbian plasticity would lead to stabilised connections between these core cells whereby a cognitive representation would form. The synaptic plasticity hypothesis has found solid experimental support although refinements appeared to be necessary [4]. The Hebbian framework has also seen many extensions and applications over the years; some reviews can be found in [3, 57, 56].

As indicated initially, one of the most appealing aspects of Cell Assemblies is their bilateral explanatory nature: On one hand assemblies aim at functional models for psychological phenomena, but on the other hand they ground these functions in the physical substrate of neurons, synapses, and networks. This congruency with the new discipline Cognitive Computation, we believe, will soon become important in the development of artificial cognitive computers: Neural hardware is currently being developed by many hardware designers worldwide, but it requires new and yet non-existing programming paradigms. At the same time neurophysiological research is growing exponentially that targets on higher cortical functions like decision making, navigation, behavioural control, or language. This research will provide the theory for the former new computers, because both should be naturally adapted to each other even though at the moment the respective research spans across disciplines. In contrast, alternative approaches like quantum-, swarm-, or molecular computing although they have their own merits do not share much similarity with human cognitive systems and therefore need other sources of inspiration to be made “smart”.

We have argued elsewhere how the framework of Hebbian cell assemblies can be extended in a very natural way by decisions, rules, and syntactic patterns [57]. “Operational Cell Assemblies” of this extended kind can in principle provide a programming paradigm for upcoming large-scale neuro-inspired hardware systems [55]. In the present work we do not repeat the respective arguments but consider a complementary aspect of assemblies that takes focus on their representational capabilities rather than computation. We will show that the internal structure of assemblies is compositional in a natural and most appealing way. Both aspects together contribute to the strength of assemblies—their potential as expressive data structures considered here and their universal computing power discussed in [55].

Many scientists loosely understand assemblies as any collection of somehow cooperating neurons. As a starting point we prefer to consider them in a more strict sense as stored patterns in an associative memory of, say, the

Willshaw or Hopfield type [2, 24, 38, 63]. Clearly this is not a definition of assemblies but more a figure of thought that captures some of Hebb’s original ideas although certainly not all. Stimuli (or other entities) are thought of being represented by patterns of neural activity across a set of neurons. In the present work—as in Hopfield and Willshaw nets in general—we assume that these patterns are just binary; neurons are either on or off (this is not a crucial simplification; it can be relaxed). Patterns are stored by means of synaptic plasticity based on pre- and post-synaptic activity. A variety of learning rules has been studied [1, 17, 18, 16, 39, 40] most of which have in common that all neurons in one assembly strengthen their mutual synapses as required by Hebb, but whether synapses get also weakened or even negative depends on the precise model variant. Because patterns will in general overlap they form overlapping blocks in auto-associative storage matrices, see Fig. 1.

The theory of associative memories mostly considers uncorrelated random patterns because this simplifies statistical analyses of stability conditions or the computation of memory capacities [2, 23, 39, 40]. In practice, overlaps between patterns are not random but reflect the similarity or relational closeness of the represented entities; they can also be seen as the ‘strength’ that connects things or thoughts associatively. The theory and models we outline below shed light on mathematical hierarchical properties of general overlap structures in assemblies.

Sub-graphs in a graph of nodes that are fully connected by pair-wise edges are called “cliques” [38]. Assemblies in auto-associative memories are cliques because each neuron lends support to every other neuron within the assembly by means of strong synapses [38]. As long as the memory matrix is not overloaded these are also “maximal cliques” meaning that every other neuron outside the assembly is less than fully connected with the neurons inside. This is necessary to make assemblies attractors of associative memory models, otherwise activity would spread from a fully activated assembly to the external neurons that receive full support as well.

Direct evidence has recently been provided for clique-codes in multiple-unit electrode recordings from rat hippocampus [21, 32, 33]. These experiments confirmed earlier indirect evidence for assemblies [19, 36]. Lin et al. [32, 33], for instance, studied startle-triggered ensemble responses in large sets of 260 simultaneously recorded CA1 neurons of rats in response to different experiences like ‘freefall inside an elevator’, ‘sudden gush of air to the back’, or ‘earthquake-like shake’. They found that the stimuli evoked diverse changes in some hippocampal neurons but could be projected by means of a multiple discriminant analysis to distinct patterns in a low-dimensional coding space. More specifically, neurons engaged in the responses were

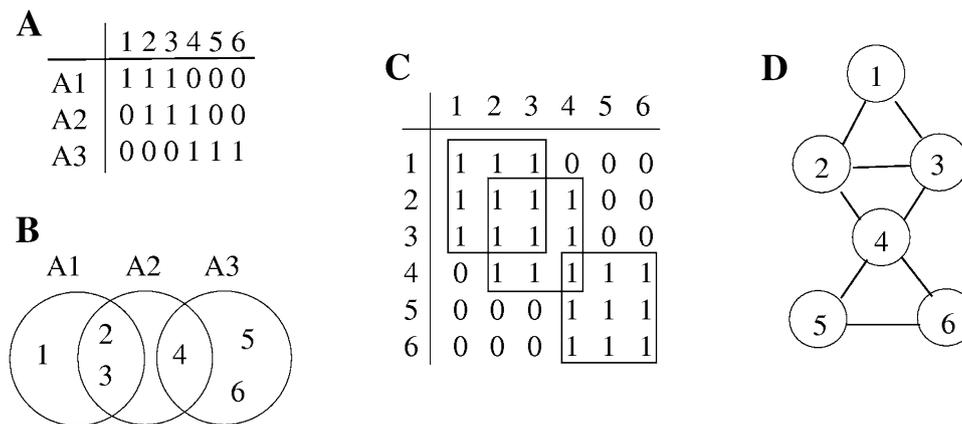


Fig. 1 Schematic of patterns and their representation as assemblies. **a** Shows 3 binary patterns A^1, A^2, A^3 with three active units ('ones') each. **b** Shows the overlap structure of the patterns as a Venn diagram. In **(c)** the patterns have been stored in an auto-associative matrix of binary synapses. The three 3×3 blocks indicate full connectivity between units in each pattern, whereby an assembly is formed; these

categorically and hierarchically organised. A broad set of cells represented general startle-responses, smaller sets were responsive to only one or two of the stimulus types (shake, gush, fall), whereas even smaller sets responded selectively to further context specialisations like the identity of one of two elevators used in the freefall paradigm. These experiments clearly demonstrate a compositional code that uses nested categories represented by large ensembles of cells such that more specific stimulus conditions are represented by smaller ensembles.

The present work argues for a complementary interpretation of hierarchical clique-codes. In addition to nested categories we suggest that assemblies may implement more complex structures named 'formal concept lattices' [20, 62]. Nested category hierarchies can be interpreted as special sub-lattices of those. Radim Bělohlávek in a so far unfortunately mostly ignored paper [5] has shown that concept lattices can be implemented in the attractor structure of bidirectional associative memories.

Formal concept analysis starts from two disjoint finite sets of 'objects' and 'features' and a relation between them: an object-feature pair is either in the relation or not. A 'formal concept' is defined as a maximal set of objects paired with the maximal set of features they share. The objects of a concept are its 'content', because informally these are the objects that the concept comprises, whereas the features are its 'intent', because they characterise the objects in terms of shared properties.

Think of a shopping basket: some things in it are red, some are vegetables, some canned. If the only red vegetables in it were tomatoes the pair of sets ($\{tomato\}, \{red, vegetable\}$) could be a valid concept in the context of the given basket. All red things in the basket could define another concept, as could all 'canned blue' things or all

are maximal blocks of connections that cannot be increased in size without adding zeros (missing connections) to the blocks. Therefore the neurons in assemblies form maximal cliques. **d** displays the graph of the network of units with **(c)** as adjacency matrix; self-connections are not shown and each edge is bidirectional. Again note that the assemblies (1,2,3), (2,3,4), and (4,5,6) form fully connected cliques

'canned blue vegetables'. Note, that formally the pair of objects and their features defines the concept, but in everyday language we often just use the features, the intent, to name it as in the previous examples. Figure 2 shows another shopping basket example.

Single objects can define concepts described by their individual collection of features if these characterise it uniquely (otherwise all objects that share the features are in the content of the concept, cf. the concept ($\{lettuce, spinach\}, \{green, vegetable\}$) in Fig. 2). Single features may similarly define concepts described by the collection of all the objects that share them. Concepts form a hierarchy, the concept lattice, with a complete partial order relation that is induced by set inclusion: the sets of objects that share some features get smaller when the respective set of features gets larger or, conversely, the more objects we choose the less features they will have in common. Precise definitions will be given in section "Formal Concepts in BAMs". There are two extremal concepts at the bottom and the top of the hierarchy. The smallest concept comprises the minimal number of objects that share all features; this can be the empty set as in the example in Fig. 2. Conversely, on the top of the hierarchy we have a concept that comprises all objects and the features they share. In our example all objects are vegetables, but especially in larger contexts usually no feature is shared by all objects, neither has any object all features. The formal concept hierarchy of a given context reflects the set-theoretic compositional relations between its feature-set and object-set.

Associative memories can implement context structures, and thereby compositional hierarchies, in a natural way. Bělohlávek [5] proved that certain bidirectional associative memories (BAMs, see [31]) can precisely implement formal concept hierarchies. We here present an

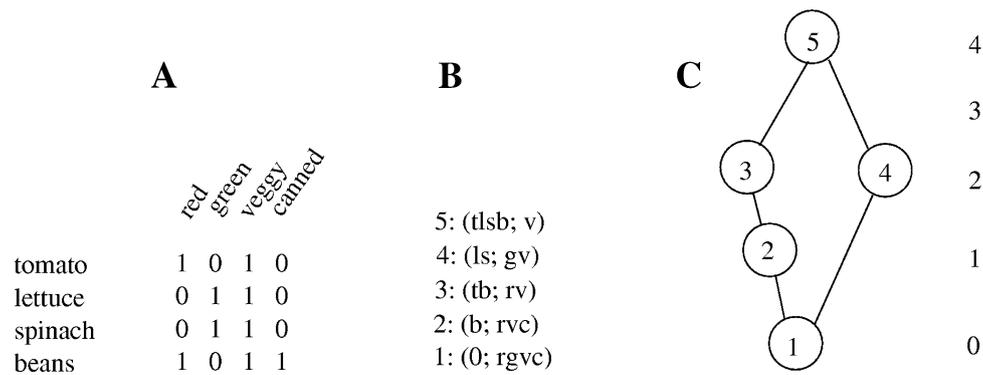


Fig. 2 Example for a simple concept lattice. **a** Shows a formal context that specifies just four objects and four features as well as the feature–object relationships between them. **b** Lists all concepts of this context, that is, all maximal blocks in **(a)** (possibly after reordering rows and columns). Note that rows (or columns) can be identical, for instance, ‘l’ & ‘s’ (initial letters abbreviate full names), and that the

size of the object sets increases from concept 1 to 5 whereas the feature sizes decrease. **c** Displays the concept lattice such that the lower concepts of connected pairs are ‘smaller’ than the higher ones, in the sense that all their objects are contained also in the higher one. The numbers on the far right indicate the cardinality of the concepts, that is, the number of objects they contain

implementation that follows Bělohlávek’s work but removes some of its less biological features. Our implementation respects Dale’s principle [9] and operates with balanced levels of excitation and inhibition. The model works with fixed non-plastic synapses between excitatory and inhibitory cells that do not need to be changed when the network learns new concepts. It is moreover enough to use just a single inhibitory neuron that acts as a global threshold control; any number can be used as long as they collectively compute the average activity of the excitatory neurons and inhibit these cells proportionally. This is in accordance with the hypothesis that interneurons may have mainly supportive functions like activity regulation but do not so much contribute directly to memory processes due to their lower number and less or non-plastic synapses. Our model furthermore can learn concept lattices incrementally by means of a Hebbian coincidence learning rule.

The section “[Formal Concepts in BAMs](#)” introduces concept lattices and Bělohlávek’s original implementation. Section “[Biological Improvements](#)” describes our attempts to make the model more biological. Afterwards implications for auto-associative memories are discussed in section “[Implications for Auto-Associative Memories](#)”. In section “[Discussion](#)” we come back to the observation of hierarchical clique-codes in the cortex and outline further possible applications of the framework of concept lattices and associative representations developed here.

Formal Concepts in BAMs

Formal Concepts

Formal concept analysis [20, 62] considers a finite set of k objects, $G = \{g_1, \dots, g_k\}$, and a finite set of l features,

$M = \{m_1, \dots, m_l\}$.¹ Objects can have a variable number of features and features can be shared by several objects. These relationships are defined by a binary relation I between G and M , such that $(g, m) \in I$ for $g \in G, m \in M$ means that object g has the feature m (and does not have it if $(g, m) \notin I$). The triple (G, M, I) defines the *context* the formal concept analysis is about (for instance, a specific shopping basket).

A (formal) concept is a pair $(A, B), A \subseteq G, B \subseteq M$ such that A is the maximal set in G that shares all features of B , and B is in turn the maximal set of features shared by all the objects in A . Formally, $A = B^\downarrow := \{g \in G | \forall m \in B : (g, m) \in I\}$ and $B = A^\uparrow := \{m \in M | \forall g \in A : (g, m) \in I\}$. The set A is said to be the *extent* of the concept and B its *intent*.

Note that both sets A and B have to be maximal. In the example in Fig. 2, $(\{beans\}, \{red, vegetable\})$ is not a formal concept because there are other red vegetables (‘tomatoes’) in the basket as well as other features in the list of features (M) that beans in the basket possess (‘canned’).

We denote by $\mathcal{B}(G, M, I)$ the set of all concepts of a given context (G, M, I) . The set of concepts can be ordered hierarchically such that it forms a complete lattice, the *formal concept lattice* [62]. More concretely: by means of the usual set inclusion we can define *sub-concepts* of ‘larger’ concepts by requiring that all objects in the smaller (sub-)concept are also objects in the larger concept; formally: for two concepts $(A_1, B_1), (A_2, B_2) \in \mathcal{B}(G, M, I)$ we put $(A_1, B_1) \leq (A_2, B_2)$ iff $A_1 \subseteq A_2$ (or, equivalently, $B_2 \subseteq B_1$). Wille [62] has shown that $\mathcal{B}(G, M, I)$ under the so-defined order-relation is a complete lattice where every set of concepts has a least upper bound (supremum or

¹ We mainly follow Bělohlávek’s notation and definitions in [5] in the present work.

direct generalisation) and a greatest lower bound (infimum or direct specialisation). The infimum of a set of concepts is the concept that contains as objects the intersection of all individual object sets, and the supremum is the concept that contains as features the intersection of all feature sets of the individual concepts. Unless there are objects that possess all features or features shared by all objects, the total maxima and minima of the hierarchy are the concepts where either the object set or the feature set is empty.

Bělohávek’s Implementation

This section summarises briefly the main mathematical results in [5], where Bělohávek has worked out that formal concept hierarchies can be implemented in the fixed point structure of bidirectional associative memories (BAM) [31]. BAMs store a set of, say P , pairs of binary 0/1-patterns $(X^p, Y^p) = (\{x_1^p, \dots, x_k^p\}, \{y_1^p, \dots, y_l^p\})$ $p = 1, \dots, P$ in the weights w_{ij} of a bi-layer neural network with dynamic equations

$$y'_j = \begin{cases} 1 & \text{for } \sum_{i=1}^k w_{ij}x_i > \theta_j^y \\ y_j & \text{for } \sum_{i=1}^k w_{ij}x_i = \theta_j^y \\ 0 & \text{for } \sum_{i=1}^k w_{ij}x_i < \theta_j^y \end{cases} \quad x'_i = \begin{cases} 1 & \text{for } \sum_{j=1}^l w_{ij}y'_j > \theta_i^x \\ x_i & \text{for } \sum_{j=1}^l w_{ij}y'_j = \theta_i^x \\ 0 & \text{for } \sum_{j=1}^l w_{ij}y'_j < \theta_i^x \end{cases} \tag{1}$$

The θ_j^y, θ_i^x are firing thresholds of the neurons. The network operates in discrete time by cycling through successive feedforward–feedback steps. Unprimed variables refer to values before a step, primed variables to values afterwards. In one feedforward–feedback retrieval cycle an input pattern x is first mapped to the second layer yielding y' and then back to the first layer to get x' . The process can apparently be iterated as desired resulting in a time-discrete dynamical system.

Whereas the goal of the classic bidirectional associative memory is to retrieve pattern pairs from partial or distorted cues [31], it is shown in [5] that they can also be used to store and retrieve concept hierarchies. In this interpretation the two network layers represent the features and objects of a formal context, respectively. One proposed learning rule for a given context (G, M, I) is

$$w_{ij} = \begin{cases} 1 & \text{if } (g_i, m_j) \in I \\ -q & \text{if } (g_i, m_j) \notin I \end{cases} \tag{2}$$

where $q = \max(k, l) + 1$. All thresholds are set equal to $-1/2$. The set of stable points of this BAM is isomorphic to the formal concept lattice induced by (G, M, I) [5]. If some feature set is used as input the corresponding objects that share all these features are found in the forward step, and

any missing features that are further shared by the objects (in addition to those already in the input) are completed in the backward step. Perhaps surprisingly, it can be shown that every finite lattice can be implemented by a BAM as above ([5], corollary 1).

A second learning rule proposed in [5] is based on ‘training’ sets $T = \{(A^p, B^p) | A^p \subseteq G, B^p \subseteq M, p \in \{1, \dots, P\}\}$. A training set is called *conceptually consistent* if it is a subset of some concept lattice. The relation table, I_T , of a conceptually consistent training set T can be constructed by

$$(g, m) \in I_T \text{ iff } \exists (A, B) \in T : g \in A, m \in B. \tag{3}$$

Bělohávek provides necessary and sufficient conditions for a training set to be conceptually consistent (Theorem 2 in [5]) and proves that conceptually consistent training sets can be stored in the sense that the concepts they comprise are fixed points of the resulting BAM using the following learning rule (Corollary 2 in [5]).

Learning rule 2: For $i = 1, \dots, k, j = 1, \dots, l$ set the thresholds $\theta_i^x = \theta_j^y = -1/2$ and weights

$$w_{ij} = \begin{cases} 1 & \text{if } \exists p \in \{1, \dots, P\} : g_i \in A^p, m_j \in B^p \\ -\max(k, l) + 1 & \text{if otherwise} \end{cases} \tag{4}$$

Figure 3 displays some conceptually consistent and inconsistent training sets for illustrative purposes. The example is based on the context in Fig. 2. The three blocks in (a) are all concepts and together (or in any smaller combination) form a conceptually consistent training set. However, this training set does not contain the full information about the context, because the concept $(\{tomato, beans\}, \{red, vegetables\})$ is not in the training set. The system after training therefore does not know of that concept (it knows that tomatoes are vegetables but not that they are red). Conversely all blocks indicated in (b) are not even concepts. Any training set that contains them is conceptually inconsistent in the sense of [5]. After training of an inconsistent training set its training examples may or

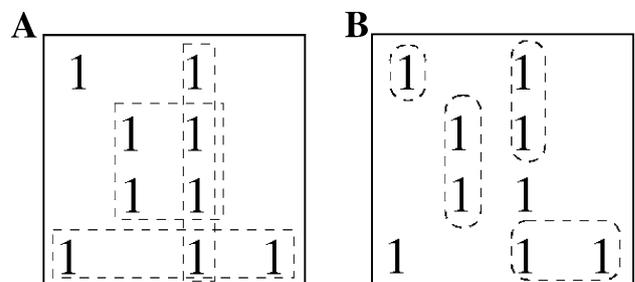


Fig. 3 Conceptually consistent and inconsistent training examples. Used in (a) and (b) is the context relation from Fig. 2. **a** Reveals some conceptually consistent training examples that all comprise maximal blocks of feature-object relations and therefore represent formal concepts of the context. **b** Shows inconsistent training sets; these also comprise blocks of positive feature-object pairs, but non-maximal blocks

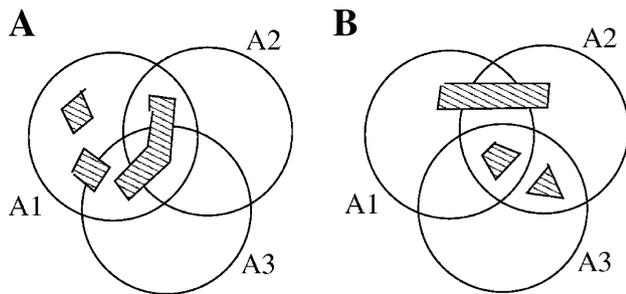


Fig. 4 Addressing of unique versus ambiguous assemblies. The circles in (a) and (b) depict a Venn diagram of the active units in three overlapping memory patterns, A^1 , A^2 , A^3 ; intersections contain units active in all the intersecting assemblies. Hatched regions depict address patterns, that is, sets of units used as starting patterns for associate retrieval. In (a) each of the three address patterns as well as any combination of them addresses pattern A^1 uniquely because the address patterns are fully contained in A^1 but in no other pattern. In (b) the hatched regions address more than a single pattern ambiguously, they are fully contained in more than one memory pattern

may not be stable fixed points of the trained BAM; the behaviour of that BAM is therefore rather undefined (Fig. 4).

Biological Improvements

Bělohávek's BAM implementation of formal concepts is provenly correct and works perfectly well as a technical system. From a biological point of view one can make some improvements outlined in this section that make the model biologically more plausible. First, note that neurons in the model have positive and negative outgoing synapses in conflict with Dale's principle that neurons in the brain express either excitatory or inhibitory synapses [9]. Secondly, observe that the inhibitory synapses in (2) or (4) are extremely large as compared to the excitatory ones, the fraction being one to the system size, $\max(k, l)$. Third, the learning rules (2) or (4) are "one-shot" rules in the sense that the whole relation table has to be known in advance for rule (2) and the whole training pattern set for rule (4) in order to compute the synaptic weights. The learning rules also require plastic inhibitory synapses.

We resolve the first and second issue by separating neurons into excitatory and inhibitory ones. More precisely, we start by assuming that the excitatory units are threshold units with specific connections controlled by global linear and unspecific inhibition per pool. The time-discrete dynamics of the networks can then be written as

$$X' = \sum_{i=1}^k x_i \quad (5)$$

$$y'_j = \Theta \left[q \sum_{i=1}^k w_{ij} x_i - cX' - \theta_j^y \right] \quad (6)$$

$$Y' = \sum_{j=1}^l y'_j \quad (7)$$

$$x'_i = \Theta \left[q \sum_{j=1}^l w_{ij} y'_j - cY' - \theta_i^x \right] \quad (8)$$

where Θ is the Heaviside function, $\Theta(x) = 1$ if $x > 0$ and 0 else.

X and Y are the potentials of the inhibitory neuron in either of the two pools; y_j , x_i are the potentials of the excitatory neurons before a simulation step; primed variables denote variables after the simulation step. θ_j^y and θ_i^x are firing thresholds, and q and c parameters of the model that balance excitation against inhibition. The thresholds and other parameters are fixed later.

In (5)–(8) excitatory neurons are separated from inhibitory neurons such that Dale's principle is satisfied. Furthermore, the synapses from excitatory cells on the inhibitory cells are all 1, such that the inhibitory neurons just sum the total activity from one pool and inhibit the excitatory neurons in the other pool proportionally. This represents an unspecific feedforward inhibition between the pools. Because, X' is computed before y' (and Y' before x') the inhibition is assumed to act (slightly) faster than the excitatory synaptic transmission.

As stated, the model assumes a single inhibitory neuron per pool. This is an extreme case that can be easily relaxed by splitting either of the inhibitory neurons arbitrarily into any number requested. These may receive input from the excitatory cells by some arbitrary but invertible synaptic matrix V . If the connections from the new inhibitory neurons to the excitatory neurons are given by the inverse of V the action of the now several inhibitory cells is effectively the same as that of the single cell before, they inhibit the excitatory cells proportional to the total activity in the second pool. This is obviously a consequence of the inhibitory neurons being linear, and it can be reached with non-negative matrix entries in V and V^{-1} only, keeping Dale's principle intact.

The rationale behind this aspect of the model is that it has been argued that the excitatory cells are the main computational work-horses in real neural systems whereas the inhibitory cells have mainly supportive functions like activation control [7]. This view is supported by the significantly smaller number of inhibitory cells and the fact that plastic (NMDA-)synapses are expressed by excitatory cells only. Therefore in our model only synapses between excitatory neurons learn, but the inhibitory neurons have fixed in-going and out-going synapses that do not need to be adapted.

Given a formal context (G, M, I) a one-shot learning rule for the excitatory synapses is

$$w_{ij} = \begin{cases} 1 & \text{if } (g_i, m_j) \in I \\ 0 & \text{if } (g_i, m_j) \notin I \end{cases} \quad (9)$$

The weights reflect exactly the membership relations of features in objects and vice versa. Furthermore, all synapses are 1 as for the inhibitory neurons. We will see below that also the weight factors c and q in the dynamic Eqs. 6 and 8 can be equal or close in order to reach proper concept retrieval. Therefore, excitation and inhibition are much more balanced than in the original implementation in section “[Bělohávek’s Implementation](#)”

To prove correctness of the model we need to show equivalence of the concepts of any formal context and the fixed points in its BAM implementation. For simplicity we do this for the choice of parameters: $1 = q = c > -\theta = .5 > 0$ and $\theta_j^y = \theta_i^x = \theta$ for all $1 \leq i \leq k, 1 \leq j \leq l$. With this choice excitatory and inhibitory synapses have the same strength and the dynamic Eqs. 5 to 8 can be simplified to

$$y'_j = \Theta \left[\sum_{i=1}^k (w_{ij} - 1)x_i - \theta \right] \tag{10}$$

$$x'_i = \Theta \left[\sum_{j=1}^l (w_{ij} - 1)y'_j - \theta \right]. \tag{11}$$

Note, that because the threshold θ is smaller than zero, an input $x = 0$ will result in all y'_j becoming active in the forward step and in turn in all x'_i becoming zero in the subsequent feedback step if “no object has all features”. This results in a fixed point that represents the correct retrieval of the total minimum of the concept hierarchy assuming that x represent the objects.

For a given context (G, M, I) and a given set of features (objects), S , we define the *coset* of S as the maximum set of objects (features) that shares the features (objects) in S . If S is a set of objects the smallest concept containing S in the formal concept lattice is then $(\text{coset}(\text{coset}(S)), \text{coset}(S))$.

If $x \in \{0, 1\}^k$ represents any object set, y' computed according to Eq. 10 will represent its coset: observe that the inhibitory contribution in $\sum_{i=1}^k (w_{ij} - 1)x_i$ in Eq. 10 counts the number of active units in x independent of the target unit j . In order to overcome the firing threshold a target unit therefore needs excitatory input from all active units in x which then just balances the inhibition, but because the threshold is negative the unit will fire. Consequently the units in the coset of x will fire but no other neurons. The same arguments hold in the feedback direction; therefore, $x' = \text{coset}(\text{coset}(x))$ and the smallest concept that comprises x is retrieved.

The above arguments imply that every concept of the concept lattice $\mathcal{B}(G, M, I)$ corresponds with a fixed point in its BAM representation and can be retrieved in a single forward–backward cycle from either its feature set or its object set. Conversely, each fixed point must be a concept because being a fixed point implies maximal connectivity between the activated features and objects; there are no

other objects that receive input from all activated features and no other features that receive input from all activated objects. Accordingly, both sets are maximal which characterises a concept of $\mathcal{B}(G, M, I)$.

Bělohávek proved for his BAM, Eq. 1 with learning rule (4), that a set of training examples $T = \{(A^p, B^p), p = 1, \dots, P\}$ is learnable in the sense that the examples become fixed points of the BAM iff T is conceptually consistent. The latter basically mean that all the pairs (A^p, B^p) must be concepts of a fixed concept lattice. The learning rule (4) is not incremental because all pattern pairs need to be known in advance in order to set the synapses.

Our implementation allows for incremental learning where only excitatory synapses get potentiated whereas the inhibitory synapses can be left entirely untouched in the course of learning.

Assume T is a conceptually consistent training set in the sense of [5]. Let $\mathcal{P} = \{T_1, \dots, T_r\}$ be any partition of T . We train T_1, \dots, T_r incrementally. Synapses after training of T_k are denoted as w_{ij}^k starting from an empty synapse matrix, $w_{ij}^0 = 0$ for all $i = 1, \dots, k; j = 1, \dots, l$. The learning rule then is

$$w_{ij}^k = \begin{cases} \max(w_{ij}^{k-1}, 1) & \text{if } \exists(A, B) \in T_k : g_i \in A, m_j \in B \\ w_{ij}^{k-1} & \text{otherwise} \end{cases}. \tag{12}$$

In each learning step this rule takes the old synaptic matrix and sets all synapses corresponding with feature–object pairs in the new pattern set to 1 (if they are not already 1). The result of incrementally learning T_1, \dots, T_s for some $0 \leq s \leq r$ is apparently the same as if the union $\cup_{i=1, \dots, s} T_i$ would have been trained using the one-shot rule (9). Therefore the rule (12) correctly stores the concepts contained in the training set up to a current learning step as well as all other concepts these contain implicitly. After all training examples have been learned, rule (12) reaches the same result as rule (9) for the context relation I_T induced by (3) and the full training set T .

Note that a special partition \mathcal{P} would be composed of the P individual pairs of the training set T , which therefore can be trained individually in sequence. Also: the order of presentation of a partition apparently does not matter for the final result as the synaptic matrix eventually reached will inevitably have collected all pairings of features and objects in the training set, no matter when they arose in the course of training.

According to the theorems in [5], after each training step the BAM will perfectly represent some formal concept lattice which contains the training pairs already shown as fixed points. There may be more fixed points, which represent additional concepts that have not been in the training sets. The represented concept lattice will grow as more and

more training examples are shown, meaning that the size of the represented lattice, that is, the number of concepts it contains, can never decrease. This is simply a consequence of the fact that a fixed point once created cannot vanish again by means of later training, because a concept must stay a concept in a conceptually consistent training set. Consequently, new concepts can be inserted into the hierarchy of the lattice, but old ones never deleted. Of course, depending on the order of training, the growth process will run through different paths of lattices.

Conceptually consistent training sets induce a relation table by means of 3. All pairings of features and objects add an entry from that table to the current synaptic matrix by learning rule 12. Each training pair must be a concept, that is, a maximal block in the context table to learn. Later training pairs must not destroy any earlier existing blocks in order for the training set to be conceptually consistent. The sequence of concept lattices represented in the time course of learning is the sequence of lattices induced by the respective blocks after each learning step.

Implications for Auto-Associative Memories

The bidirectional associative memory separates objects and features into two distinct layers. Auto-associative memories, in contrast, comprise only a single layer. Nonetheless, the considerations of the previous sections can also provide some insight into the retrieval properties of auto-associative memories.

We consider auto-associative memories of the Willshaw-type with k binary neurons and again a single globally acting inhibitory neuron that provides a dynamic threshold control:

$$X' = \sum_{i=1}^k x_i \quad (13)$$

$$x'_j = \Theta \left[q \sum_{i=1}^k w_{ij} x_i - cX' - \theta_j^x \right]. \quad (14)$$

The network stores a set of P binary 0/1-patterns $T = \{A^p | A^p = (a_1^p, \dots, a_k^p)^T \in \{0, 1\}^k\}$ by means of the clipped Hebbian coincidence rule

$$w_{ij} = \min(1, \sum_{p=1}^k a_i^p a_j^p). \quad (15)$$

This rule is in effect the same as (9) if the input and output patterns there are identical. It can easily be serialised as in (12) leading to an incremental algorithm.

In the auto-associative case fully connected blocks in the coupling matrix define cliques in the sense of graph theory if the connection matrix is interpreted as an adjacency

matrix between some set of nodes. These blocks correspond at the same time with formal concepts of some context (G, G, W) where G is the set of entities the neurons in our model code for,² and I is equivalent to the coupling matrix defined by (15).

Classical associative memory theory aims at making all single patterns and (whenever possible) only those fixed points of the network dynamics. It has been observed previously that especially at high memory load “mixture” patterns and an exponential number of additional fixed points appear in some models [2, 23], these, however, were considered retrieval errors. Formal concept theory instead suggests that they may be related to the concept structure inherent in the overlaps of the memory patterns. This is an area that allows for and needs future research. It may especially provide insight into how generalising categories are formed if associative memories become overloaded.

By construction and previous arguments an address pattern x will after a single step of (14) excite its coset, $x' = \text{coset}(x)$, that is, the set of all units that receive input from all units in x . These units cannot itself be fully contained in a bigger pattern; therefore, under iteration of the dynamics a sequence of ever growing patterns is impossible. To the contrary, under further iteration of $x' = \text{coset}(x)$ we have $\text{coset}(\text{coset}(x')) = x'$, which leaves only two possibilities: either x' is a fixed point or x' and $\text{coset}(x')$ form a limit cycle of length two.

We say a pattern x addresses one of the stored patterns in $T = \{A^p | A^p = (a_1^p, \dots, a_k^p) \in \{0, 1\}^k \text{ for } p = 1, \dots, P\}$, say A^1 , if $x \subseteq A^1$, or, informally, if all active ones in x are also active in the memory pattern. It is then clear that the two cases above, fixed point or two-cycle, correspond with the case where the pattern x either addresses a single memory pattern uniquely in which case this pattern is retrieved after one step as a fixed point, or x addresses more than one pattern in which case $x' = \text{coset}(x)$ is the union of the addressed memory patterns (more precisely their active units), and $\text{coset}(x')$ is their intersection. The associative memory therefore oscillates between a state where all units in all addressed memory patterns are jointly active and a state where only the units common in all addressed memory patterns are active. For a single uniquely addressed pattern both states are the same.

Discussion

In summary we have shown that Bělohávek’s BAM implementation of formal concept lattices [5] can be nicely improved towards more biological realism by separated

² These entities may be features or objects or just anything, we do not make such distinctions in the auto-associative setup.

classes of excitatory and inhibitory neurons and incremental learning of excitatory neurons only. A spiking neuron implementation is also possible which will be described elsewhere. We have only presented very simple toy-examples in the present work. However, the proper functioning of the model was shown for arbitrary formal contexts. For additional examples and a data-base application see [5, 45, 64]. Retrieval in auto-associative memories has further been related to formal concept hierarchies as well with an interesting new result regarding the meaning of retrieval errors. Important future work seems possible in that direction if the assumption is relaxed that only memory patterns can be meaningful attractors.

Assemblies in auto-associative memories can be considered cliques in the strict sense of graph theory because in our interpretation of an assembly all its neurons are mutually connected but no other neuron has full connectivity to the assembly. In the BAM setup, concepts are rather “bi-cliques” where two disjoint sets of units are mutually fully connected; nonetheless, each set still lends full support to itself indirectly via the bi-synaptic pathway through the opposite pool.

The bidirectional associative memory distinguishes feature and objects layers. Thereby it naturally supports cascaded hierarchies where one or more lower level layers feed bidirectionally into higher level layers to build super-concepts. We hypothesise that this might be the case in the sensory hierarchies of the mammalian brain.

Numerous attempts explain visual or auditory processing as a hierarchy of successively more complex and increasingly invariant categorisers, e.g., [8, 12, 15, 34, 48]. Strong experimental evidence has been provided to support this view [27, 43, 53, 54] up to the highest cognitive levels in human [44, 42]. Similarly, many scientists considered feedback processes, often interpreted as top-down attention [26, 47, 49] or more recently the action of some generative models that aim to reconstruct or predict the input [12, 46]. The framework of formal concepts adds interesting new aspects to these studies worthy to explore further in the future.

Note that there are several specific differences between hierarchical categorisers and the concept-storing BAMs considered here. Both models have a feedforward phase where lower order features are integrated (in some sense) by higher order units such that a set of features can excite a set of objects. In the framework of formal concepts the object units that become excited are exactly all the units that comprise all active features in the input. In categoriser networks the activated object units are less well defined; they may be excited only by some input features and if so then only to a more or less extent. These differences refer in part to the discrete 0/1-nature of the BAM unit activities and synapses, but also to the special threshold strategy of the

concept-BAM which requires that all active input units match with non-zero synapses into any target neuron that has to be activated. A second major difference between categoriser hierarchies and concept-lattice BAMs is that the latter add feedback to the system which aims at completing all features that the initially excited objects share. This set is the intersection of the full feature sets of the excited units as represented by their non-zero synapses. Some categoriser hierarchies with feedback have been described, for example, the classical neo-cognitron [13]. In these networks the feedback rather projects the synaptic fields of the excited object neurons back into the feature space whereby something like the superposition or union of the synaptic fields/feature sets is reconstructed but not their intersection. A third difference is that lateral inhibition is often used to perform some winner-take-all-type selection of the maximally excited object neuron in categoriser networks; in the BAM an arbitrary number of units in any layer can be active. If a winner is selected the back-projected synaptic fields can be used to reconstruct missing parts in the input pattern, see, e.g. [13, 14]. Apparently, the precise similarities and differences between hierarchical categorisers and concept-BAMs need further investigation. A unifying theory may be possible, perhaps in a Bayesian framework.

Lin et al.[33], based on multiple-electrode recordings in behaving rats [21, 32], argued for categorical representations of stimulus conditions related to behavioural responses in hippocampal neurons which comprise nested sets of neurons that get smaller the more specific the stimulus conditions are. These results to some degree parallel the expectations from concept lattices but consider only the mapping of features to one output aspect, the reflexive response. “Objects” in the sense of formal concept analysis in the cited or similar extended experiments may characterise the rat’s responses as a combination of aspects of the possible responses. It is our suggestion that the pyramidal hierarchical structure depicted in [33] (Fig. 1) therefore may represent only part of a fuller relational network; it may perhaps reflect an ordered path of the full underlying lattice of stimulus–response concepts. A more complete picture could be obtained by adding more possible reactions (objects) that have some aspects in common but differ with regard to others and study how more complete stimulus–reaction sets are reflected in the firing of the recorded neurons.

Tsunoda et al.[54] presented another example where concept hierarchies may play a crucial role (cf., also [53, 27]). In combined optical and single unit recordings from inferotemporal cortex of Macaques they found that visual objects evoke blobs of activity in the optical recordings that stay roughly constant in cortical location and size if complex stimuli get successively simplified. Objects with some shared features evoked patterns with some blobs in

common. In some cases existing blobs disappeared, in other cases new blobs appeared when a stimulus was simplified. Simultaneously recorded single unit activity from neurons within and outside blobs was largely consistent with the optical response patterns. These experiments suggest categorical responses of neurons in blob areas such that neurons are sensitive to certain stimulus properties which, however, can be shared between representations of different entities. The appearance and disappearance of blobs when stimuli are simplified may reflect the structure of feature–object relationships implemented in the cortical network. Data in [27, 43, 11] may be interpreted in a similar vein.

Experimenters are usually bound to the observation of activity of neurons only, but not their connections. Collectively elevated firing rates or synchronous spiking therefore define cliques based on mutual correlations only. Whether those result from auto-associative or hetero-associative physical connections (or both, or any other scheme) can in general not be decided. Our results suggest that the observable dynamic response patterns as well as their compositional structure should be different for either the bidirectional or the auto-associative scheme: auto-associative memories do not support a full retrieval of the concept lattice defined by a set of patterns, whereas the bidirectional memory does. Auto-associative connections also were shown to lead to oscillations between the union and intersection of all addressed memory patterns in case of ambiguities, whereas such type of response would contradict pure concept retrieval. It cannot be excluded, however, that modifications of auto-associative memories can retrieve and represent arbitrary concepts defined by their memory patterns. It is also likely that the cortex uses a mix of auto-associative and bidirectional processes (and possibly more).

We have introduced Hebb’s cell assemblies as a theory that ultimately aims at integrating Physiology and Psychology. If assemblies are implemented in the brain and do reflect concept processing as proposed here, this should correlate with the formation and processing of psychological categories and feature–object relations. Psychologists have investigated the nature of categorical and conceptual processes in humans, e.g. [10, 28, 35, 37, 51], however, without final conclusions yet [52]. Formal concept analysis may suggest further crucial experiments. From the cognitive psychological perspective it would seem appealing that concept lattices reveal a natural compositional structure that maps collections of features to object sets, which in turn can be integrated into super-concepts and so on in a hierarchical way. This reflects the compositional structure of most cognitive domains.

Hebbian cell assemblies, we believe, will soon have multiple technical applications in forth-coming future computer systems where those are based on simulated

neurons or distinct neural hardware. One reason for this is that they are “by construction” adapted to this hardware and there is currently no alternative programming paradigm available for neural cognitive large-scale simulations (only for biophysical simulations). The present work outlined some aspects of the beautiful representational structure of assemblies, whereas in earlier work we have extended the framework towards rules, language, and procedural processes in general [57, 55, 58]. This should give the framework of cell assemblies a good position in approaching the challenge of future truly cognitive computing.

Acknowledgement The author acknowledges support by the Engineering and Physical Sciences Research Council (grant EP/C010841/1, COLAMN—A Novel Computing Architecture for Cognitive Systems based on the Laminar Microcircuitry of the Neocortex) and the European Community (FACETS—Fast Analog Computing with Emergent Transient States). Comments by the reviewer are also greatly acknowledged.

References

1. Abbott LF, Nelson SB. Synaptic plasticity: taming the beast. *Nat Neurosci.* 2000;3:1178–83.
2. Amit DJ. *Modeling brain function.* Cambridge: Cambridge University Press; 1988.
3. Amit DJ. The Hebbian paradigm reintegrated: local reverberations as internal representations. *Behav Brain Sci.* 1995;18:617–57.
4. Bi G-q, Poo M-m. Synaptic modification by correlated activity: Hebb’s postulate revisited. *Annu Rev Neurosci.* 2001;24:139–66.
5. Bělohlávek R. Representation of concept lattices by bidirectional associative memories. *Neural Comput.* 2000;12:2279–90.
6. Braitenberg V. Cell assemblies in the cerebral cortex. In: Heim R, Palm G, editors. *Theoretical approaches to complex systems.* Berlin: Springer; 1978. p. 171–88.
7. Braitenberg V, Schüz A. *Anatomy of the cortex.* Berlin: Springer-Verlag; 1991.
8. Dayan P. Images, frames, and connectionist hierarchies. *Neural Comput.* 2006;18:2293–319.
9. Eccles JC, Fatt P, Koketsu K. Cholinergic and inhibitory synapses in a pathway from motor-axon collaterals to motoneurons. *J Physiol (Lond).* 1954;126:52462.
10. Feldman J. Minimization of boolean complexity in human concept learning. *Nature.* 2000;407:630–3.
11. Freedman DJ, Riesenhuber M, Poggio T, Miller EK. A comparison of primate prefrontal and inferior temporal cortices during visual categorization. *J Neurosci.* 2003;23:5235–46.
12. Friston K. Hierarchical models in the brain. *PLoS Comput Biol.* 2008;4:e1000211.
13. Fukushima K. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol Cybern.* 1980;36:193–202.
14. Fukushima K. Restoring partly occluded patterns: a neural network model. *Neural Netw.* 2005;18:33–43.
15. Fukushima K, Myaki S. Neocognitron: a new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognit* 1982;15:455–69.
16. Fusi S, Abbott LF. Limits on the memory storage capacity of bounded synapses. *Nat Neurosci.* 2007;10:485–93.
17. Fusi S, Annunziato M, Badoni D, Salamon A, Amit DJ. Spike-driven synaptic plasticity: theory, simulation, VLSI implementation. *Neural Comput.* 2000;12:2227–58.

18. Fusi S, Drew PJ, Abbott LF. Cascade models of synaptically stored memories. *Neuron*. 2005;45:599–611.
19. Fuster JM. *Memory in the cerebral cortex*. An empirical approach to neural networks in the human and nonhuman primate. Cambridge, MA: MIT Press; 1994.
20. Ganter B, Wille R. *Formal concept analysis: mathematical foundations*. Berlin: Springer-Verlag; 1999.
21. Harris KD, Csicsvari J, Hirase H, Dragoi G, Buzsaki G. Organization of cell assemblies in the hippocampus. *Nature*. 2003;424:552–5.
22. Hebb D. *The organization of behavior*. New York: Wiley; 1949.
23. Hertz J, Krogh A, Palmer RG. *Introduction to the theory of neural computation*. Redwood city: Addison Wesley; 1991.
24. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. *PNAS*. 1982;79:2554–8.
25. Indiveri G. Synaptic plasticity and spike-based computation in VLSI networks of integrate-and-fire neurons. *Neural Inf Process—Lett Rev*. 2007;11:135–46.
26. Itti L, Koch C, Niebur E (1998) A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20:1254–1259
27. Kiani R, Esteky H, Mirpour K, Tanaka K. Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *J Neurophysiol*. 2007;97(6):4296–309.
28. Kirk U. The neural basis of object-context relationships on aesthetic judgment. *PLoS ONE*. 2008;3:e3754.
29. Knoblauch A, Markert H, Palm GG. An associative cortical model of language understanding and action planning. In: Mira J, Alvarez JR, editors. *Proceedings of IWINAC 2005, First International Work-Conference on the Interplay between natural and artificial computation, Las Palmas de Gran Canaria, Spain*. vol. 3562, *Lecture notes in computer science*. Berlin, New York: Springer; 2005. p. 405–14.
30. Knoblauch A, Kupper R, Gewaltig M-O, Körner U, Körner E. A cell assembly based model for the cortical microcircuitry. *Neurocomputing*. 2007;70:1838–42.
31. Kosko B. Bidirectional associative memory. *IEEE Trans Syst Man Cybern*. 1988;18:49–60.
32. Lin L, Osan R, Shoham S, Jin W, Zuo W, Tsien JZ. Identification of network-level coding units for real-time representation of episodic experiences in the hippocampus. *Proc Natl Acad Sci USA*. 2005;102:6125–30.
33. Lin L, Osan R, Tsien JZ. Organizing principles of real-time memory encoding: neural clique assemblies and universal neural codes. *Trends Neurosci*. 2006;29:48–57.
34. Logothetis NK, Sheinberg DL. Visual object recognition. *Annu Rev Neurosci*. 1996;19:577–621.
35. Martinovic J, Gruber T, Müller MM. Coding of visual object features and feature conjunctions in the human brain. *PLoS One*. 2008;3:3781.
36. Miyashita Y, Chang HS. Neural correlate of pictorial short term memory. *Nature*. 1988;331:68–70.
37. Nosofsky RM, Gluck MA, Palmeri TJ, McKinley SC, Clautier P. Comparing models of rule-based classification learning: a replication and extension of Shepard, Hovland and Jenkins (1961). *Mem Cognit*. 1994;22:352–69.
38. Palm G. *Neural assemblies: an alternative approach to artificial intelligence*. Berlin: Springer-Verlag; 1982.
39. Palm G. Memory capacities of local rules for synaptic modification. *Concepts Neurosci*. 1991;2:97–128.
40. Palm G, Sommer F. Information capacity in recurrent McCulloch-Pitts networks with sparsely coded memory states. *Network*. 1992;3:177ff.
41. Pulvermüller F. Constituents of a neurological theory of language. *Concepts Neurosci*. 1992;3:157–200.
42. Quiroga RQ, Reddy L, Kreiman G, Koch C, Fried I. Invariant visual representation by single neurons in the human brain. *Nature*. 2005;435:1102–7.
43. Quiroga RQ, Kreimann G, Koch C, Fried I. Sparse but not 'grandmother-cell' coding in the medial temporal lobe. *Trends Cogn Sci*. 2008(a);12:87–91.
44. Quiroga RQ, Mukamel R, Malach EA, Fried I. Human single-neuron responses at the threshold of conscious recognition. *Proc Natl Acad Sci USA*. 2008(b);105:3599–604.
45. Rajapakse R, Denham M. Fast access to concepts in concept lattices via bidirectional associative memories. *Neural Comput*. 2005;17:2291–300.
46. Rao RPN, Ballard DH. Dynamic model of visual recognition predicts neural response properties in the visual cortex. *Neural Comput*. 1997;9:721–63.
47. Raz A, Buhle J. Typologies of attentional networks. *Nat Rev Neurosci*. 2006;7:367–79.
48. Riesenhuber M, Poggio T. Hierarchical models of object recognition in cortex. *Nat Neurosci*. 1999;2(11):1019–25.
49. Saalman YB, Pigarev IN, Vidyasagar TR. Neural mechanisms of visual attention: how top-down feedback highlights relevant locations. *Science*. 2007;316:1612–5.
50. Schemmel J, Meier K, Mueller E. A new VLSI model of neural microcircuits including spike time dependent plasticity. In: *Proceedings IJCNN*. IEEE Press; 2007. p. 1711–6.
51. Shepard R, Hovland CL, Jenkins HM. Learning and memorization of classifications. *Psychol Monogr Gen Appl*. 1961;75:1–42.
52. Sloman SA, Rips LJ. *Similarity and symbols in human thinking*. Cambridge, MA: MIT Press; 1998.
53. Tanaka K. Neuronal mechanisms of object recognition. *Science*. 1993;262:685–8.
54. Tsunoda K, Yamane Y, Nishizaki M, Tanifuji M. Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns. *Nat Neurosci*. 2001;4:832–8.
55. Wennekers T. Operational cell assemblies as a paradigm for brain-inspired future computing architectures. *Neural Inf Process—Lett Rev*. 2006;10:135–45.
56. Wennekers T, Palm G. Cell assemblies, associative memory, and temporal structure in brain signals. In: Miller R, editors. *Time and the brain*. vol. 3, *CABR—conceptual advances in brain research*. Harwood Academic Publishers; 2000. p. 251–73.
57. Wennekers T, Palm G. Modelling generic cognitive functions with operational Hebbian cell assemblies. In: Weiss ML, editors. *Neural network research horizons*. Nova Science Publishers; 2007. p. 225–94.
58. Wennekers T, Garagnani M, Pulvermüller F. Language models based on Hebbian cell assemblies. *J Neurosci (Paris)*. 2006;100:16–30.
59. Wickelgren WA. Context-sensitive coding, associative memory, and serial order in (speech) behavior. *Psychol Rev*. 1969;76:1–15.
60. Wickelgren WA. Webs, cell assemblies, and chunking in neural nets. *Concepts Neurosci*. 1992;3:1–53.
61. Wijekoon JHB, Dudek P. Compact silicon neuron circuit with spiking and bursting behaviour. *Neural Netw*. 2008;21:524–34.
62. Wille R. Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival I, editor. *Ordered sets*. Dordrecht: Reidel; 1982. p. 445–70.
63. Willshaw DJ, Buneman OP, Longuet-Higgins HC. Non-holographic associative memory. *Nature*. 1969;222:960–2.
64. Wolff KE. A first course in formal concept analysis. In: Faulbaum F, editors. *StatSoft '93 Advances in statistical software*. Gustav Fischer Verlag; 1993. p. 429–38.
65. Yamashita Y, Tani J. Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment. *PLoS Comput Biol* 2008;4:e1000220