Neural Computing and Applications
https://doi.org/10.1007/s00521-019-04670-3

ORIGINAL ARTICLE

# A biologically plausible network model for pattern storage and recall inspired by Dentate Gyrus

V. Vidya Janarthanam[1] · S. Vishwanath[1] · A. P. Shanthi[1]

Received: 12 October 2018 / Accepted: 6 December 2019


**Abstract**
In the race to achieve better performance, artificial intelligence has become more about the end rather than the means, which is general intelligence. This work aims to bridge the gap between the two by finding a complementary midline. The objective of this work is to project the role of Dentate Gyrus in enhancing the performance of an autoassociative network, paving the way to develop a biologically plausible neural network which, in the future, would help in simulating the network present in our brain. The proposed network imbibes biological similarities with respect to connectivity, weight updation, and activation function. Dentate Gyrus uses pre-integration lateral inhibition form of learning, and the autoassociative network is implemented using Hopfield network. The performance of the autoassociative network in the presence and absence of Dentate Gyrus is observed across multiple parameters. The results show an increase of 38% in storage capacity and a decrease of 15% in the error tolerance capability of the network in the presence of Dentate Gyrus.

**Keywords** CA3 · Memory · Hippocampus · Hopfield network · DG

## 1 Introduction

Artificial general intelligence has remained a holy grail for AI researchers, while working of the neural network in our brain has remained a mystery for neuroscience researchers worldwide. With breakthroughs in the field of neuroscience, neuroimaging, and cognitive science, a better picture has emerged. Harnessing these findings and the increase in computation power, this work proposes the first step toward building a network functionally similar to the brain in silico.

Any intelligent system, artificial or biological, requires a knowledge base, and the purpose of building a knowledge base is fulfilled only when it stores and retrieves data effectively and efficiently. If we want to achieve human-level intelligence in the future, we need to structure and process knowledge like humans do. So, the question is where do we start? The answer lies in the region known as hippocampus in our brain, which plays a major role in memory storage and retrieval. It can be considered as a code book for the knowledge stored in our brain, and it is also responsible for creating the code book. Hippocampus is a part of the archicortex, the oldest region of the brain. Being one of the earliest regions to develop, it can be said that hippocampus is the starting point of the evolutionary road leading to memory and intelligence as we know it today.

The role of hippocampus in memory has been an active area of research for a long time now. The hippocampal region includes the Dentate Gyrus (DG), hippocampus proper, and the subiculum (Sub). Hippocampal proper comprises four subfields CA1, CA2, CA3 and CA4, where CA stands for Cornu Ammonis (horn of the Egyptian god Amun). A number of models have been developed to represent the different functions of the hippocampus [16, 30, 35, 40]. The hippocampal circuit along with its adjacent regions is shown in Fig. 1. The entorhinal cortex (EC) has multiple layers. Layers II and III are a part of the input pathway from the higher regions of the brain. Layer V is part of the output pathway. The numbers given in the figure represent the number of neurons in each region of the hippocampus as given in [31, 39]. The dashed line represents backprojection of information into the higher

✉ V. Vidya Janarthanam
vidya_v@cs.annauniv.edu; vidushi.meenu@gmail.com

[1] Department of Computer Science and Engineering, College of Engineering, Guindy, Anna University, Chennai, India

Published online: 28 January 2020

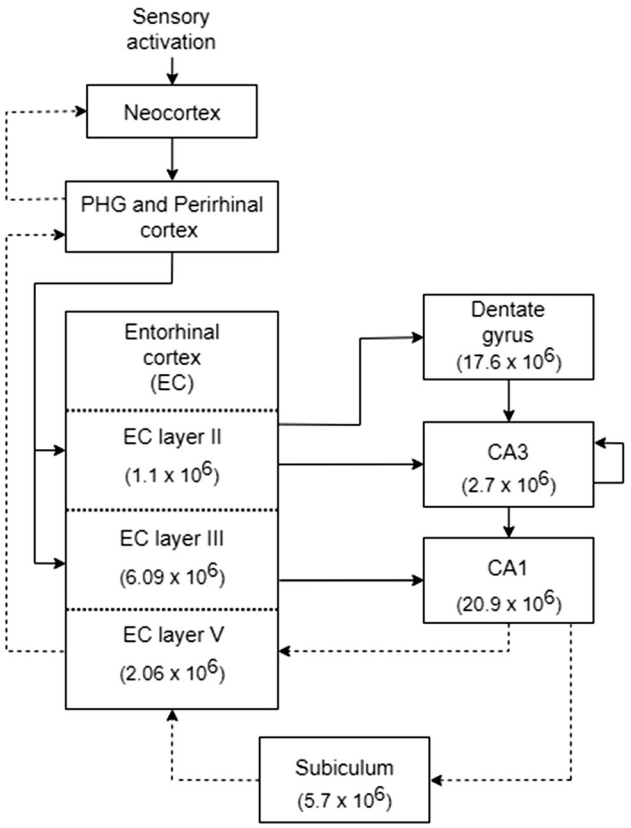


**Fig. 1** Medial temporal lobe with number of neurons in each of the hippocampal regions

regions of the brain after being processed by the hippocampus.

Pattern completion and pattern separation are two important mechanisms which enable the hippocampus in retrieving episodic memory [27, 29]. Pattern completion happens when the whole memory is retrieved by completing the received cues. The trigger could be anything, a location, an object, a reward, or an experience. Pattern completion can occur from any part of the memory, that is, the cue could be anything related to the memory. This is different from pattern association where retrieval happens only in one direction when presented with a stimulus [27].

When there is some correlation between patterns, the capacity of a network shrinks [16]. In order to overcome this, the hippocampus performs pattern separation. It basically makes sure that two similar events can be differentiated during retrieval. There are different subprocesses which help the hippocampus in achieving pattern separation, and the Dentate Gyrus plays a significant role as listed in [29].

This paper aims to develop a functionally similar model of a subregion of the hippocampus known as Dentate Gyrus, which performs certain basic functionalities similar to its biological counterpart. Section 2 describes the network components in detail, starting from how a neuron is represented in a network to the working of an autoassociative network. Section 3 presents the proposed work, elaborating on implementation details like weight updation rule, activation function, and the structure of the network. Section 4 describes in brief the system setup and constraints placed on the proposed network. Section 5 explains the various evaluation methods used and the results obtained.

## 2 Network components and properties

A neuron can be considered as the building block of a nervous system, and the connection between two neurons is called a synapse [4]. Together, multiple neurons and synapses form a network which enables humans to process information obtained from the world around. The strength of a synapse plays a major role in learning and memory. The change in synaptic strength between the neurons as a result of activation in the respective neurons is called synaptic plasticity. This section describes the type of neural coding used to encode information and elaborates on the functions of Dentate Gyrus, which helps in improving the efficiency of CA3.

### 2.1 Neural coding

In humans, the activity pattern in a layer of neurons, which arise as a result of stimulation of the neurons, represents an item [25]. The term item could refer to something complex like an object, a location, a word, a song, or it could refer to something more fundamental which leads up to the complex item, like an edge, a musical note, an alphabet, or even just a dot.

Complexity of the stimulus that a neuron responds to depends upon its position in the neural pathway. For example, in case of visual stimulus, rod and cones are the sensory neurons which bring about phototransduction. This signal is passed on via bipolar cells to the retinal ganglion cell (RGC). The RGCs have a center-surround receptive field which are of two types—on-center, off-surround and off-center, on-surround, which are stimulated when the respective areas are activated [11]. Basically, it responds to a dot of light or darkness. On the other end, a neuron in the V4 area of the visual cortex responds to color, depth, shape, and motion information [26]. Further higher in the pathway, V4 is strongly connected to the inferior temporal cortex which performs the task of the object, face recognition which goes on to connect to the hippocampus. All along the pathway information is represented as a pattern of active neurons in each layer.

When a neuron fires, it is said to generate a spike, and when it fires repeatedly it generates a spike train. Analogous to information coding in computers, a spike can be treated as '1' and silence as '0'. As a result, a spike train becomes a sequence of 0s and 1s. The number of spikes generated by a neuron per unit time is the neuron's firing rate.

Different coding schemes have been proposed to encode information from the activity of neurons. In the case of temporal coding, spike timing is considered to hold information. In the rate coding model, the firing rate of a neuron is said to encode information. Population coding represents information in a distributed form. Each neuron on its own does not contain any indispensable information, but when combined with responses of many other neurons, that is, a population of neurons, it holds a significant amount of information. Basically, a set of features, which bring about a response in their respective neuron, are used to represent a particular item. The neighboring neurons in a population are tuned to respond to inputs, which are very similar, but not the same, leading to redundancy.

The problem of redundancy in population coding can be controlled if only one neuron, of the population representing similar features, responds to a stimulus. This type of coding is called sparse coding, since only a small subset of all the available neurons fire at any particular time to represent a particular item. Compared to population coding, less energy is consumed in case of sparse coding. The sparseness is measured in terms of number of active neurons in a given population of neurons. Increase in sparseness of the representation leads to increase in capacity of the network of neurons [1, 21, 22]. Higher storage capacity and lower power consumption are desirable features for designing a network. There is enough evidence to support the biological significance of sparse coding in sensory systems responsible for sight, sound, smell, and touch [6, 10, 14, 21, 38]. Taking all these factors into consideration, the network proposed in this work uses sparse coding to encode information.

## 2.2 Dentate Gyrus

Dentate Gyrus is the first region in the hippocampal formation. The Dentate Gyrus receives input from the ECII region via perforant path, and it projects to the CA3 region of the hippocampus via mossy fibers. The subprocesses carried out by the Dentate Gyrus region are vital for CA3 to function efficiently.

The Dentate Gyrus plays a compelling role in the formation of new memories. A significant increase in the firing of the dentate granule cells is observed in the case of a subject being presented with a novel stimulus [19]. This surge in firing is attributed to the rise in level of dopamine in the hippocampus [24, 42]. The novelty detection process though is carried out by the CA1 region, and the result is passed on to the ventral tegmental area (VTA) which is responsible for controlling the release of dopamine [15]. Since CA1 region is not implemented in this work, whether the CA3 network has to perform storage or retrieval is set manually.

The mossy fiber connection to CA3 region is sparse and strong, and this plays a vital role in encoding new information [34]. It has to be noted that a single mossy fiber plays a significant role in firing a neuron, but the mossy fiber pathway as a whole has a little say in the activation of CA3 cells due to sparse activation in the Dentate Gyrus [37].

Dentate Gyrus is said to perform pattern separation, while encoding, as a result of different subprocesses [18, 27]. The number of granule cells in the Dentate Gyrus region is almost 16 and 6.5 times the neuron count in ECII and CA3 region, respectively; thus, there is a significant scale-up in the number of neurons [31, 39]. Basically, this means that the activation pattern in ECII is projected onto Dentate Gyrus, giving rise to a sparse activation, given the higher count of neurons in the latter. This expansion recoding helps in decorrelating the input patterns and improves input variability, leading to pattern separation [3, 27]. The expansion recoding process can be realized in silico by using competitive learning to remap the activation pattern in ECII onto the higher-dimensional Dentate Gyrus layer [28].

## 2.3 CA3

The CA3 region of the hippocampus is composed of pyramidal neurons. A pyramidal neuron is a multipolar neuron, that is, it has many dendrites and one axon. The dendrites and axon have many branches which enable the neuron to connect to a number of neurons. As the number of branches increases, the ability of a neuron to integrate and distribute information increases. The CA3 pyramidal neurons receive input from the Dentate Gyrus via the mossy fiber pathway, from the entorhinal cortex via the perforant path and from CA3 pyramidal neurons itself via recurrent collaterals. The recurrent collaterals terminate at the apical dendrites, and the mossy fibers terminate at the basal dendrites [41]. The recurrent collaterals enable the CA3 region to perform pattern completion, and there is no self connection between neurons in the CA3 network. The mossy fiber synapses are large, and also powerful since they terminate closer to the cell body, which means they play a relatively significant role in firing of a CA3 neuron. Also, the firing in the Dentate Gyrus region is very sparse and this promotes pattern separation in CA3, which is necessary for the CA3 network to perform efficiently [34].

The CA3–CA3 connections form an autoassociative network, explained in Sect. 2.4, similar to a Hopfield network [2, 17]. The difference is that the Hopfield network is fully connected, whereas the CA3–CA3 network has diluted connectivity. The dilution in biological networks is due to the biological upper limit on the number of inputs a neuron can receive. Some modifications were made to the Hopfield network to accommodate this difference, while simulating the CA3–CA3 autoassociation network [30]. This dilution in connectivity leads to an increase in storage capacity of the CA3–CA3 network.

### 2.4 Autoassociative memory

There are two types of association: *heteroassociation*, where the input pattern retrieves the output pattern, and also the patterns are different from each other, and *autoassociation*, where the input pattern acts as a cue to retrieve the output pattern, and the patterns are similar [23]. As mentioned in Sect. 2.3, CA3 performs autoassociation. When presented with a degraded or incomplete pattern, the CA3 returns the original pattern. For example, when we see a cat in any angle, we recognize that it is a cat, which is autoassociation. Associating a cat with the word 'cat' is called heteroassociation.

Major part of this work deals with improving the performance of an autoassociation network. Figure 2 depicts the working of an autoassociative network. Let A be a pattern which has already been stored in the network and $\{a_1, a_6, a_8, a_{12}, a_{24}\}$ be the neurons representing the subpatterns constituting A. The neuron $a_i$ is activated when sum of all incoming action potentials into $a_i$ causes a drop in the neuron's membrane potential leading to the firing of $a_i$. Similarly, the output neuron A fires when all the neurons representing the subpatterns of A fire. In Fig. 2, an autoassociative network is given with 25 interconnected neurons $\{a_0 \ldots\ldots a_{24}\}$. The neurons $a_i$ and $a_j$ are interconnected with the condition that $i \neq j$. Given an input pattern which is an incomplete version of pattern A, the network, via recurrent collaterals, will complete the patterns, leading to the original pattern A. The number of cycles required by the biological network to complete a pattern is estimated to be between 4 and 6 [16, 40].

Let us assume that neurons $\{a_1, a_6, a_{12}\}$ fire as in Fig. 2(i). The output of the current state leads to the firing of other neurons in the network, as a result of the action potential added on via the recurrent collateral. This continues for a number of cycles, as mentioned above, until eventually all neurons representing pattern A are activated.

The capacity of an autoassociative memory depends on the number of neurons in the network, relationship between the patterns stored, activation ratio, and dilution in connectivity. While simulating an autoassociative network, the vectors representing the patterns should be mutually orthogonal. In such cases, $N - 1$ vectors can be stored, where $N$ is the number of neurons [8]. In high-dimensional systems, finding a set of orthogonal vectors is a computationally expensive process. If the patterns are sparse, it is safe to say that the vectors representing it are nearly orthogonal, which makes more sense from a biological perspective. In the case of biological networks, patterns tend to overlap when there is some similarity between stimuli.

## 3 Proposed work

This work does not deal with modeling the behavior of a biological neuron. Instead, it focuses on modeling some of the higher-level functions or behavior of a biological neural network like, neural layers with excitatory and inhibitory neurons, pattern completion, and pattern separation. The

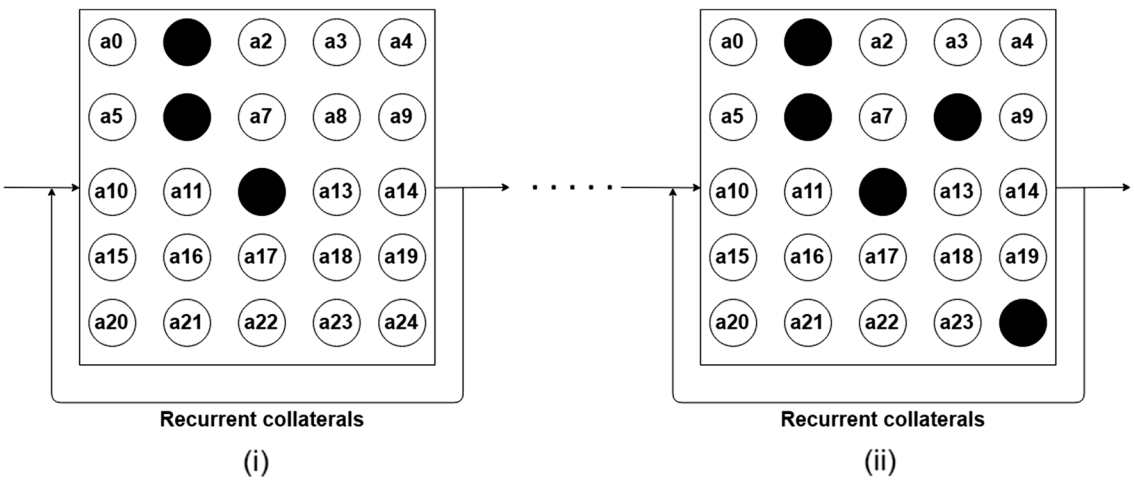


**Fig. 2** Pattern completion performed by an autoassociative network

idea is that mimicking the biological process behind learning and memory will take us a step closer to achieving human-level intelligence. Accordingly, storage and retrieval process in this paper follows the working of the hippocampal region.

From a computer science perspective, Fig. 3 is a neural network with varying weight update rules and activation functions. The proposed network neither has any mechanism to backpropagate errors to previous states, nor does have a uniform weight update rule or activation function calculation, which is the case in most of the traditional artificial neural networks. In this context, this work proposes a heterogeneous neural network capable of performing pattern storage and retrieval.

From a biological perspective, Fig. 3 represents a subset of Fig. 1, which is an architecture of the hippocampus. The proposed network in this context tries to find a midline between the operations of a biological and artificial neural network. A simulation of the hippocampus region exists, as given in [29]. The proposed work deviates from the existing implementation in various aspects. The major difference is that the existing network works with rate coding, while the proposed network works with sparse coding mechanism. This essentially means that the existing network is analog, and the proposed network is discrete and binary.

This work simulates the hippocampus region, sans CA1. The theory that Dentate Gyrus enables the CA3 network to separate, store, and retrieve similar patterns is put to test. In Fig. 3, the equation besides an arrow represents a weight update rule and the equation within the box represents an activation function. The subsequent sections describe the methods used to realize the network.

### 3.1 The role of Dentate Gyrus

As mentioned in Sect. 2.2, the Dentate Gyrus region uses expansion recoding and competitive learning techniques to bring about pattern separation. A biologically plausible form of competitive learning is implemented using pre-integration lateral inhibition in the proposed network. This method incorporates the functions performed by an inhibitory interneuron in a biological neural network [32]. Pre-integration is preferred over post-integration, because it can represent multiple, overlapping patterns and also it adapts to a coding scheme, local or distributed, based on the input received [33]. Equation 1 defines the activation function of neurons in the Dentate Gyrus layer as derived in [32].

$$h_j^{\mathrm{DG}} = \sum_{i=1}^{m} w_{ij}^{\mathrm{EC\_DG}} x_i \left(1 - \alpha \max_{\substack{k=1 \\ k \neq j}}^{n} \left\{ \frac{w_{ik}^{\mathrm{EC\_DG}}}{\max_{l=1}^{m} \{ w_{lk}^{\mathrm{EC\_DG}} \}} \frac{y_k}{\max_{l=1}^{n} \{ y_l \}} \right\} \right)^{+} \tag{1}$$

The above equation accounts for the effect of inhibitory interneurons. $h_j^{\mathrm{DG}}$ is the activation received by neuron $j$ in the Dentate Gyrus layer. Figure 4 has been included to give a better understanding of Eq. 1. In this equation, $x$ and $y$ represent the neuron in ECII and the Dentate Gyrus, respectively. The term $\sum_{i=1}^{m} w_{ij}^{EC\_DG} x_i$ represents the total input activation received by neuron $j$ from ECII, and $\alpha$ controls the magnitude of lateral inhibition applied to the neuron.

$$\triangle w_{ij}^{\mathrm{EC\_DG}} = \beta \frac{(x_i - \overline{x})}{\sum_{k=1}^{m} x_k} \frac{(y_j - \overline{y})}{\sum_{k=1}^{n} y_k} \tag{2}$$

Calculation of synaptic weight $w_{ij}^{\mathrm{EC\_DG}}$ between ECII $(i)$ and Dentate Gyrus $(j)$ is given by Eq. 2. The weight between presynaptic neuron $i$ and postsynaptic neuron $j$ is given by $w_{ij}^{EC\_DG}$. $\beta$ is used to set the learning rate. $\overline{x}$ and $\overline{y}$

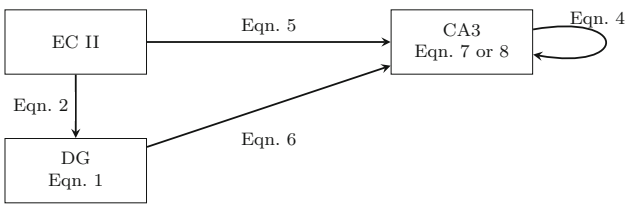


**Fig. 3** Proposed network

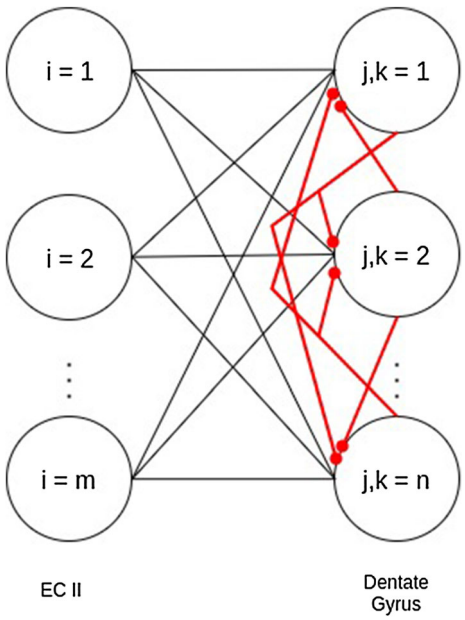


**Fig. 4** Competitive network comprising EC II and Dentate Gyrus regions. The connections in red represent the inhibition incorporated into the network (color figure online)

are the mean activation values of the layer $x$ and $y$, respectively.

### 3.2 CA3 using Hopfield network

As mentioned in Sect. 2.3, the CA3 network is similar to an autoassociative network. To make it similar to the actual CA3 region in the hippocampus, some changes and additions have to be made. The CA3 network in this work is implemented using a Hopfield network. The Hopfield network operates in two different modes. The first mode is operational when a new memory is given as an input. In this case, additional activation is received by a CA3 neuron from the Dentate Gyrus region along with the input from the ECII region. This additional activation leads to modification in the weights of the recurrent connections in the Hopfield network. In the second mode, the CA3 network is given an incomplete or incorrect version of a pattern, which has already been stored in the network. In this case, there is no additional activation from the Dentate Gyrus, but the recurrent connections drive the network to retrieve the complete original pattern. In short, mode one performs storage using pattern separation and mode two performs retrieval using pattern completion. The storage capacity of a Hopfield network is given as $0.0138N$ [9]. This value is achieved when the probability of error in the input pattern is $<0.01$.

In general, hippocampal models use rate coding to represent information. In such cases, weights are updated based on the average firing rate of a neuron [13, 30]. Hebbian learning and its variations, like covariance rule, BCM rule, Oja's rule to name a few, following rate coding are used extensively to model biological neural networks. In case of sparse coding, we assume that information is encoded in the pattern of activation of neurons and not in the firing rate of a neuron. The equation for weight update while using rate coding and sparse coding varies accordingly.

In the case of recurrent collaterals, synapses interconnect neurons within the same layer. Equation 3 used in [30] performs weight updation using Hebbian covariance rule, where $N$ is the number of neurons in the layer, $P$ is the total number of patterns learned, and $w_{ij}$ is the synaptic weight between neurons $i$ and $j$. $\eta_i^\mu$ and $\eta_j^\mu$ represent the firing rate of neurons $i$ and $j$ in pattern $\mu$. The value of $a$ is based on the firing rate $\eta$ and defined as $a = \langle\eta\rangle^2/\langle\eta\rangle^2$. Since sparse coding does not consider firing rate, the equation is modified to 4, as given in [20].

$$w_{ij} = \frac{1}{\mathrm{Na}^2}\sum_{\mu=1}^{P}(\eta_i^\mu - a)\left(\eta_j^\mu - a\right) \tag{3}$$

$$w_{ij}^{\mathrm{RC}} = \frac{1}{N}\sum_{\mu=1}^{P}(x_i^\mu - a)\left(x_j^\mu - a\right) \tag{4}$$

Equations 3 and 4 look alike but differ in two aspects. First, $\eta$ represents the firing rate of a neuron, whereas $x$ depicts the state of a neuron given by $x = \pm 1$, where $+1$ implies the neuron is active and $-1$ dormant. Second, the calculation of $a$ in Eq. 4 is different. It is the mean of the vector $x^\mu$, $a = E[x_i^\mu]$.

It is proposed that synapses between EC II and CA3 are said to exhibit associative modification, that is, they follow Hebbian learning [5]. The synaptic weight between the two regions is given by Eq. 5, derived from Oja's rule. In the equation, $\triangle w_{ij}^{\mathrm{EC\_CA3}}$ represents the change in weight being calculated. The term $w_{ij}^{\mathrm{EC\_CA3}}$ represents the current weight between EC II and CA3. The term $y_j$ is the state of the $j$th neuron in CA3, and $\gamma$ is the learning rate.

$$\triangle w_{ij}^{\mathrm{EC\_CA3}} = \gamma y_j\left(x_i - y_j w_{ij}^{\mathrm{EC\_CA3}}\right) \tag{5}$$

The mossy fiber synapses on the other hand do not show signs of long-term potentiation (LTP) or long-term depression (LTD) [29]. This was later proved to be false, and it was established that the mossy fiber synapse onto CA3 exhibited both Hebbian and non-Hebbian forms of LTP and LTD [12, 36]. In non-Hebbian form of potentiation, the synaptic strength is modified as a result of activation of presynaptic or postsynaptic neuron, but not both, which is the case in Hebbian. Non-Hebbian potentiation is thought to bring about Hebbian plasticity eventually. In the proposed network, only Hebbian plasticity has been implemented, due to the lack of clarity about the functions of non-Hebbian plasticity.

Therefore, the weight update for the mossy fiber synapse also follows Oja's rule, given by Eq. 6, where $\triangle w_{ij}^{\mathrm{DG\_CA3}}$ represents the change in weight being calculated. The term $w_{ij}^{\mathrm{DG\_CA3}}$ represents the current weight between DG and CA3. The term $y_j$ is the state of the $j$th neuron in CA3, and $\gamma$ is the learning rate.

$$\triangle w_{ij}^{\mathrm{DG\_CA3}} = \gamma y_j\left(x_i - y_j w_{ij}^{\mathrm{DG\_CA3}}\right) \tag{6}$$

The following two equations calculate the activation received by a neuron in CA3. In the equations, the terms $x$, $y$, $z$ represent a neuron in the ECII, DG, and CA3 regions, respectively. While encoding patterns, the total activation received by neuron $i$ in CA3 is given by Eq. 7. The first part of the equation calculates the activation received from ECII. The second part of the equation calculates the activation received from Dentate Gyrus. Since the activation via recurrent collateral is suppressed during encoding, it is not represented in the calculation.

$$h_i^{\mathrm{CA3}} = \sum_{j=1}^{m} w_{ji}^{\mathrm{EC\_CA3}} x_j + \sum_{k=1}^{n} w_{ki}^{\mathrm{DG\_CA3}} y_k. \tag{7}$$

During recall, the total activation received by neuron $i$ in CA3 is given by Eq. 8. The first part of the equation calculates the activation received from ECII. The second part of the equation calculates the activation received via the recurrent collaterals. Since the activation via Dentate Gyrus is suppressed during recall, it is not represented in the calculation.

$$h_i^{\mathrm{CA3}} = \sum_{j=1}^{m} w_{ji}^{\mathrm{EC\_CA3}} x_j + \sum_{l=1}^{r} w_{li}^{\mathrm{RC}} z_l. \tag{8}$$

## 4 Simulation details

The network was implemented on a system powered by an Intel Xeon processor, with 32 GB RAM and NVIDIA Titan X (Maxwell architecture, 12 GB). The number of neurons in each layer was set in proportion to the biological count mentioned in Sect. 2.2 and scaled down. Biologically, the number of synapses per neuron varies with neuronal density, but we did not take this into consideration and all the layers were fully connected [7].

The network weights between two regions were initialized to equal values, summing up to 1. Even after the weights were modified by the algorithm, they were normalized to sum up to 1. Table 1 gives the list of notations used in the algorithms and their description. It is to be noted that some changes have been made in the representation of the equations. This change is to accommodate different equations in a single algorithm. The components of the equation remain the same as described in the previous sections.

The description of Algorithm 1, used to store patterns, is as follows. For each pattern $p_i$ received by ECII, the corresponding neurons in $N_{\mathrm{ECII}}$ are activated. The pattern is then processed via two pathways, ECII-CA3 and ECII-DG-CA3. The final activation is the sum of activations received via the two pathways.

In the ECII–DG–CA3 pathway, the *while* loop continues to execute till $w^{\mathrm{EC\_DG}}$ reaches a stable state, that is, till it stops changing between iterations. For each neuron $j$ in DG, the activation function is calculated. After calculating the activation value for all neurons in DG, if a neuron is activated, the weights corresponding to the activated neuron are updated. If a neuron is not active, the weights remain unchanged.

The ECII–CA3 pathway is much simpler. For each neuron $u_{\mathrm{e}}$ in ECII, the activation received via the perforant path using Eq. 5 is calculated. Then, it is added to the activation received via the mossy fiber, as given in Eq. 7. Once $h^{\mathrm{CA3}}$ has been calculated for a pattern, the weights of the network are updated as mentioned in Algorithm 1. The algorithm, basically, stores patterns in the network by modifying the connection weights accordingly and returns the modified weights.

**Table 1** Notations used in the algorithm and their description

| Symbol | Description |
|---|---|
| $P$ | Patterns to be stored in the network |
| $N_{\mathrm{DG}}$ | Set of $n$ neurons in DG |
| $N_{\mathrm{ECII}}$ | Set of $m$ neurons in ECII |
| $N_{\mathrm{CA3}}$ | Set of $r$ neurons in CA3 |
| $u_{\mathrm{d}}$ | A neuron in DG |
| $u_{\mathrm{e}}$ | A neuron in ECII |
| $u_{\mathrm{c}}$ | A neuron in CA3 |
| $w^{\mathrm{EC\_DG}}$ | Synaptic weights between ECII and DG, initialized to $1/mn$ |
| $w^{\mathrm{RC}}$ | Synaptic weights of recurrent collaterals, initialized to $1/r^2$ |
| $w^{\mathrm{DG\_CA3}}$ | Synaptic weights between DG and CA3, initialized to $1/nr$ |
| $w^{\mathrm{EC\_CA3}}$ | Synaptic weights between ECII and CA3, initialized to $1/mr$ |
| $h_j^{\mathrm{DG}}$ | Activation function of neuron j in DG |
| $h_r^{\mathrm{CA3}}$ | Activation function of neuron i in CA3 |
| $\alpha$ | Parameter to control inhibition |
| $\beta$ | Learning rate |
| $\gamma$ | Learning rate in Oja's rule |

**Algorithm 1:** Pattern storage

**Input** : $P$- Set of Patterns $\{p_1, p_2 \dots p_m\}$, $N_{DG}$=$\{u_d^1, u_d^2 \dots u_d^n\}$, $N_{CA3}$=$\{u_c^1, u_c^2 \dots u_c^r\}$, $N_{ECII}$=$\{u_e^1, u_e^2 \dots u_e^m\}$, $w^{EC\_DG}$, $w^{RC}$, $w^{DG\_CA3}$, $w^{EC\_CA3}$, $h_j^{DG}$, $\alpha$, $\beta$, $\gamma$

**Output:** $w^{EC\_DG}$, $w^{RC}$, $w^{DG\_CA3}$, $w^{EC\_CA3}$

**foreach** $p_i \in P$ **do**
  **Incoming activation:** $N_{ECII} \leftarrow p_i$
  **while** $w_{ij}^{EC\_DG}$ *changes between iterations* **do**
    **foreach** $u_d^j \in N_{DG}$ **do**
      $h_j^{DG} =$
      $$\sum_{i=1}^{m} w_{ij}^{EC\_DG} u_e^i \left(1 - \alpha \max_{\substack{k=1 \\ k \neq j}}^{n} \left\{ \frac{w_{ik}^{EC\_DG}}{\max_{l=1}^{m}\{w_{lk}^{EC\_DG}\}} \frac{u_d^k}{\max_{l=1}^{n}\{u_d^l\}} \right\}\right)^{+}$$
    **end**
    **foreach** $u_d^j \in N_{DG}$ **do**
      **if** $u_d^j = 1$ **then**
        Update weight using $\triangle w_{ij}^{EC\_DG} = \beta \frac{\left(u_e^i - \overline{u_e}\right)}{\sum_{k=1}^{m} u_e^k} \frac{\left(u_d^j - \overline{u_d}\right)}{\sum_{k=1}^{n} u_d^k}$
      **end**
      **else**
        No weight updation
      **end**
    **end**
  **end**
  **foreach** $u_c^i \in N_{CA3}$ **do**
    $h_i^{CA3} = \sum_{j=1}^{m} w_{ji}^{EC\_CA3} u_e^j + \sum_{k=1}^{n} w_{ki}^{DG\_CA3} u_d^k$
  **end**
  Update weights:
  $\triangle w_{ij}^{RC} = \frac{1}{r}(u_c^i - a)(u_c^j - a)$
  $\triangle w_{ij}^{EC\_CA3} = \gamma u_c^j \left(u_e^i - u_c^j w_{ij}^{EC\_CA3}\right)$
  $\triangle w_{ij}^{DG\_CA3} = \gamma u_c^j \left(u_d^i - y_c^j w_{ij}^{DG\_CA3}\right)$
**end**
**return** $\left(w^{EC\_DG}, w^{RC}, w^{DG\_CA3}, w^{EC\_CA3}\right)$ ; `// Modified weights`

During recall, given in Algorithm 2, the network weights are set as received from Algorithm 1. A pattern $p'_k$ is given as an input to the network by activating the corresponding neurons in $N_{\text{ECII}}$. Pattern $p'_k$ is an incomplete or corrupt version of a pattern $p_k \in P$. The recall algorithm is set to go on for five cycles, an average of the number of cycles mentioned in [16, 40]. For each neuron $i$ in CA3, the activation received is calculated using Eq. 8. The outcome of this algorithm is, either, a complete and correct $p_k$, or, an incorrect pattern.

**Algorithm 2:** Pattern recall

**Input** : $p_k$'- Modified version of $p_k$
**Output:** $N_{CA3}$ - Final activation in CA3
**Incoming activation**: $N_{ECII} \leftarrow p'_k$
**while** $(t \leq 5)$ **do**
  **foreach** $u_c^i \in N_{CA3}$ **do**
    $h_i^{CA3} = \sum_{j=1}^{m} w_{ji}^{EC\_CA3} x_j + \sum_{l=1}^{r} w_{li}^{RC} z_l$
  **end**
**end**
**return** $(N_{CA3})$ ;

## 5 Evaluation

The significance of Dentate Gyrus in improving the performance of a CA3 network is established by the following evaluation methods. The two architectures being compared here are a stand-alone CA3 network, referred as without DG, and a CA3 network connected to DG, referred as with DG. The stand-alone CA3 network here is nothing but a Hopfield network. Comparing these two architectures helps in inferring the effect of Dentate Gyrus.

The input patterns used for evaluation were generated randomly. The length of the input pattern corresponds to the number of neurons in the input layer of the architecture being used. Striving for a fair comparison between the two architectures, the same input patterns were presented to them both. Therefore, the number of neurons in the ECII region and the CA3 region was set equal for evaluation purposes.

The values for the learning parameters $\alpha$ and $\beta$ were set as 0.25 and 1, respectively. It was based on the values

given in [32]. The value for $\gamma$ was set at 0.30 after a number of trials.

### 5.1 Storage capacity

Storage capacity gives the number of patterns that can be stored in the network before it begins to malfunction, that is, the network is no longer able to retrieve the whole pattern, when presented with an incorrect or incomplete pattern. In Fig. 5, $x$-axis represents the size of the network in terms of the number of neurons in CA3 region. The $y$-axis represents the number of patterns that can be stored in the network before retrieval becomes faulty. The rest of the network is scaled accordingly.

It is evident from the graph that in the presence of Dentate Gyrus, the capacity of the network is higher. On an average, $\approx 0.137N$ and $\approx 0.189N$ patterns can be stored in a network of $N$ neurons, in the absence and presence of Dentate Gyrus, respectively, that is, a 38% increase in the capacity of the network. To give an idea of what it means in a large-scale network, a total of $\approx 510{,}300$ and $\approx 361{,}800$ patterns can be stored in a CA3 network with $1.7 \times 10^7$ neurons, in the presence and absence of Dentate Gyrus, respectively.

### 5.2 Pattern separation

To project the role of Dentate Gyrus in performing pattern separation, patterns with different levels of overlap were stored in a network of 100 neurons. The capacity of the network was then calculated. The overlap in patterns was calculated using Hamming distance.

As it is clear from Fig. 6, in the presence of Dentate Gyrus, the capacity of the network to store overlapping patterns is much higher. For example, when the average overlap between patterns stored in the network is set at 25%, a capacity of approximately $0.15N$ and $0.08N$ is achieved in the presence and absence of Dentate Gyrus, respectively. It is evident from the graph that the performance of the network degrades with an increase in overlap.

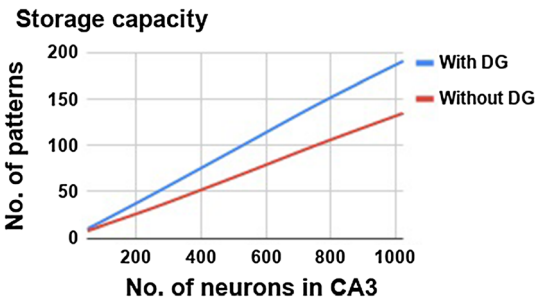


**Fig. 5** Number of patterns that can be stored effectively, given the size of a network in terms of the number of neurons in the CA3 region

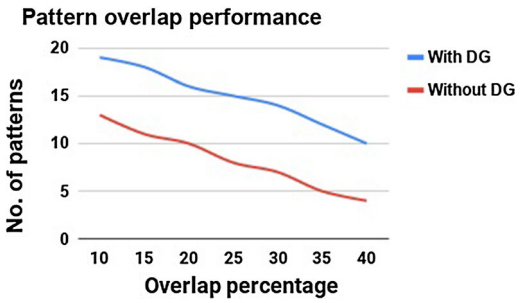


**Fig. 6** Variation in the number of patterns that can be stored against the percentage of overlap between the input patterns

However, in the presence of DG, the performance of the network is better.

This increase in capacity can be attributed to the fact that during pattern separation random neurons are activated in the CA3 region, in addition to the activation brought about by the perforant path. These random activations help in pulling apart similar patterns before storing them in the network. This helps in the retrieval of memories, which are similar, but not the same.

### 5.3 Fault tolerance

Fault tolerance is the ability of a network to deal with errors in the input pattern. The original version of these input patterns has already been stored in the network. The error could be due to incomplete pattern presentation, or due to change in few bits of the pattern. In this case, the ability of the proposed network to handle errors in the input pattern, while eventually retrieving the original pattern, is tested.

The tests were conducted by varying the number of patterns stored in the network. For example, 19 is the maximum number of patterns that can be stored in a network of 100 neurons, in the presence of Dentate Gyrus. The network is said to be working at 100% capacity if 19 patterns are stored, and it is at approximately 50% capacity if 9 patterns are stored.

From Fig. 7, it can be inferred that fault tolerance capacity of the network with Dentate Gyrus is on an

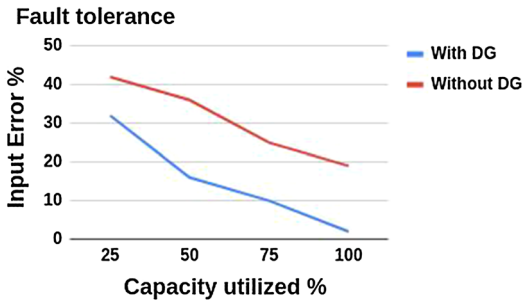


**Fig. 7** Error tolerance capability of the network at different capacity levels

average 15% lower. This setback in fault tolerance can be attributed to the increase in number of active neurons required to represent a pattern. In the presence of Dentate Gyrus, additional neurons are activated in the CA3 region by the mossy fiber pathway. The sparseness of the patterns stored in the CA3 region, or the Hopfield network to be more specific, has reduced as a result of the additional activation. Also, the additional activation means that the incoming pattern is modified before storage. This does not mean that the memory is modified, it is the representation of the memory that is modified. This modification, while retrieving, adds to the error in the input pattern, thereby bringing down the overall level of error tolerance.

The above results prove that better performance can be achieved by incorporating biologically plausible variations in a neural network. The evidence is pretty clear that the Dentate Gyrus region plays a major role in improving the performance of an autoassociative network. The storage capacity of the network has increased significantly. The proposed network is able to store and retrieve patterns which have some degree of overlap, which is necessary in case of biological patterns.

The ability of the proposed network to work with overlapping patterns comes with a trade-off. The probability of error in the network needs to be at a minimum for the network to store more patterns. This, however, means that the network is able to generalize highly similar patterns. The sparseness of patterns indicates that more patterns can be stored and retrieved with less energy. This will prove to be useful while building a hardware equivalent of the network.

## 6 Conclusion

Designing a network which is functionally similar to the network in our brain is the ultimate way to understand how the brain works and also to achieve human-level intelligence in machines. As a promising step toward achieving this goal, the results obtained are in accord with the inference provided by experiments conducted on actual brain tissue from the hippocampus region. An increase of 38% in storage capacity is achieved in the presence of Dentate Gyrus. The error tolerance capability of the network decreases by 15%, while trying to achieve better performance when overlapping patterns are involved.

The network is a fully connected one and can be further improved by adapting more biologically feasible features. For example, the synapse-to-neuron ratio can be varied across layers, network dilution can be set at different levels based on the actual biological network, and the computation of weight and activation can be parallelized in order to realize a network of higher magnitude. Due to the lack of clarity in the current literature, designing a prototype which is an actual replica of the biological neural network is not possible in the near future. This work hopes to start off as an initial step toward achieving that goal.

**Acknowledgements** We gratefully acknowledge the support of NVIDIA Corporation with the donation of the TITAN X GPU used for this research.

## Compliance with ethical standards

**Conflict of interest** The authors of this manuscript state that there is no conflict of interests.

## References

1. Amari S (1989) Characteristics of sparsely encoded associative memory. Neural Netw 2(6):451–457. https://doi.org/10.1016/0893-6080(89)90043-9
2. Amit DJ (1989) Modeling brain function—the world of attractor neural networks. Cambridge University Press, New York
3. Babadi B, Sompolinsky H (2014) Sparseness and expansion in sensory representations. Neuron 83(5):1213–1226
4. Byrne J, Dafny N (1997) Neuroscience online: an electronic textbook for the neurosciences. Department of Neurobiology and Anatomy, The University of Texas Medical School at Houston. http://nba.uth.tmc.edu/neuroscience/. Accessed Jan 2017
5. Chattarji S, Stanton PK, Sejnowski TJ (1989) Commissural synapses, but not mossy fiber synapses, in hippocampal field CA3 exhibit associative long-term potentiation and depression. Brain Res 495(1):145–150
6. Crochet S, Poulet JF, Kremer Y, Petersen CC (2011) Synaptic mechanisms underlying sparse coding of active touch. Neuron 69(6):1160–1175
7. Cullen DK, Gilroy ME, Irons HR, LaPlaca MC (2010) Synapse-to-neuron ratio is inversely related to neuronal density in mature neuronal cultures. Brain Res 1359:44–55
8. Fausett L (1994) Fundamentals of neural networks: architectures, algorithms, and applications. Prentice-Hall, Inc, Upper Saddle River
9. Hertz J, Krogh A, Palmer RG (1991) Introduction to the theory of neural computation. Addison-Wesley/Addison Wesley Longman, Reading
10. Honegger KS, Campbell RA, Turner GC (2011) Cellular-resolution population imaging reveals robust sparse coding in the drosophila mushroom body. J Neurosci 31(33):11772–11785
11. Hubel DH, Wiesel TN (1959) Receptive fields of single neurones in the cat's striate cortex. J Physiol 148(3):574–591
12. Jaffe D, Johnston D (1990) Induction of long-term potentiation at hippocampal mossy-fiber synapses follows a hebbian rule. J Neurophysiol 64(3):948–960
13. Levy WB, Hocking AB, Wu X (2005) Interpreting hippocampal function as recoding and forecasting. Neural Netw 18(9):1242–1264
14. Lin AC, Bygrave AM, De Calignon A, Lee T, Miesenböck G (2014) Sparse, decorrelated odor coding in the mushroom body enhances learned odor discrimination. Nat Neurosci 17(4):559
15. Lisman JE, Grace AA (2005) The hippocampal-VTA loop: controlling the entry of information into long-term memory. Neuron 46(5):703–713. https://doi.org/10.1016/j.neuron.2005.05.002

16. Marr D (1971) Simple memory: a theory for archicortex. Philos Trans R Soc Lond Ser B 262:23–81
17. Menezes R, Monteiro L (2011) Synaptic compensation on hopfield network: implications for memory rehabilitation. Neural Comput Appl 20(5):753–757
18. Neunuebel JP, Knierim JJ (2014) CA3 retrieves coherent representations from degraded input: direct evidence for CA3 pattern completion and Dentate Gyrus pattern separation. Neuron 81(2):416–427
19. Nitz D, McNaughton B (2004) Differential modulation of CA1 and Dentate Gyrus interneurons during exploration of novel environments. J Neurophysiol 91(2):863–872
20. Okada M (1996) Notions of associative memory and sparse coding. Neural Netw 9(8):1429–1458
21. Olshausen BA, Field DJ (2004) Sparse coding of sensory inputs. Curr Opin Neurobiol 14(4):481–487
22. Palm G (1989) On the asymptotic information storage capacity of neural networks. In: Eckmiller R, v.d. Malsburg C (eds) Neural computers, vol 41. Springer, Berlin, Heidelberg, pp 271–280
23. Palm G (2013) Neural associative memories and sparse coding. Neural Netw 37:165–171
24. Prince LY, Bacon TJ, Tigaret CM, Mellor JR (2016) Neuromodulation of the feedforward dentate GYRUS-CA3 microcircuit. Front Synaptic Neurosci 8:32. https://doi.org/10.3389/fnsyn.2016.00032
25. Quiroga RQ, Panzeri S (2013) Principles of neural coding. CRC Press, Boca Raton
26. Roe AW, Chelazzi L, Connor CE, Conway BR, Fujita I, Gallant JL, Lu H, Vanduffel W (2012) Toward a unified theory of visual area V4. Neuron 74(1):12–29
27. Rolls E (2013) The mechanisms for pattern completion and pattern separation in the hippocampus. Front Syst Neurosci 7:74
28. Rolls ET (2016a) Cerebral cortex: principles of operation. Oxford University Press, Oxford
29. Rolls ET (2016b) Pattern separation, completion, and categorisation in the hippocampus and neocortex. Neurobiol Learn Mem 129:4–28
30. Rolls ET, Treves A, Foster D, Perez-Vicente C (1997) Simulation studies of the CA3 hippocampal subfield modelled as an attractor neural network. Neural Netw 10(9):1559–1569
31. Šimić G, Kostović I, Winblad B, Bogdanović N (1997) Volume and number of neurons of the human hippocampal formation in normal aging and alzheimer's disease. J Comp Neurol 379(4):482–494
32. Spratling MW, Johnson M (2002) Preintegration lateral inhibition enhances unsupervised learning. Neural Comput 14(9):2157–2179
33. Spratling MW, Johnson M (2004) Neural coding strategies and mechanisms of competition. Cognit Syst Res 5(2):93–117
34. Treves A, Rolls ET (1992) Computational constraints suggest the need for two distinct input systems to the hippocampal CA3 network. Hippocampus 2(2):189–199
35. Tung W, Quek C (2007) A brain-inspired fuzzy semantic memory model for learning and reasoning with uncertainty. Neural Comput Appl 16(6):559–569
36. Urban NN, Barrionuevo G (1996) Induction of hebbian and non-hebbian mossy fiber long-term potentiation by distinct patterns of high-frequency stimulation. J Neurosci 16(13):4293–4299
37. Urban NN, Henze DA, Barrionuevo G (2001) Revisiting the role of the hippocampal mossy fiber synapse. Hippocampus 11(4):408–417
38. Vinje WE, Gallant JL (2000) Sparse coding and decorrelation in primary visual cortex during natural vision. Science 287(5456):1273–1276
39. West MJ, Slomianka L (1998) Total number of neurons in the layers of the human entorhinal cortex. Hippocampus 8(1):69–82
40. Willshaw DJ, Buckingham J (1990) An assessment of marr's theory of the hippocampus as a temporary memory store. Philos Trans R Soc Lond B 329(1253):205–215
41. Witter MP (2007) Intrinsic and extrinsic wiring of CA3: indications for connectional heterogeneity. Learn Mem 14(11):705–713
42. Yang K, Broussard JI, Levine AT, Jenson D, Arenkiel BR, Dani JA (2017) Dopamine receptor activity participates in hippocampal synaptic plasticity associated with novel object recognition. Eur J Neurosci 45(1):138–146. https://doi.org/10.1111/ejn.13406