



OPEN

A non-memory-based functional neural framework for animal caching behavior

Sharon Mordechay² & Oren Forkosh^{1,2}✉

The brain's extraordinary abilities are often attributed to its capacity to learn and adapt. But memory has its limitations, especially when faced with tasks such as retrieving thousands of food items—a common behavior in scatter-hoarding animals. Here, we propose a brain mechanism that may facilitate caching and retrieval behaviors, with a focus on hippocampal spatial cells. Rather than memorizing the locations of their caches, as previously hypothesized, we suggest that cache-hoarding animals employ a static mechanism akin to hash functions commonly used in computing. Our mathematical model aligns with the activity of hippocampal spatial cells, which respond to an animal's positional attention. We know that the region that activates each spatial cell remains consistent across subsequent visits to the same area but not between areas. This remapping, combined with the uniqueness of cognitive maps, produces persistent hash functions that can serve both food caching and retrieval. We present a simple neural network architecture that can generate such a probabilistic hash that is unique to the animal and not sensitive to environmental changes. This mechanism could serve a virtually boundless capacity for the encoding of any structured data.

Animals have much to hide in order to survive. Some species evade potential predators or prey by finding cover, using camouflage¹, mimicry, or other means of disguise². Others conceal their eggs or offspring, mask illnesses or injuries to avoid being targeted by predators³, or stash valuable resources such as food.

Scatter hoarding is probably the largest-scale manifestation of secretive behavior in the animal kingdom. Many animal species engage in this behavior, which involves storing food at multiple cache sites to preserve it for times when food is scarce⁴. Several bird species, such as the Siberian tit (*Poecile cinctus*), were claimed to cache several tens of thousands of food items per individual in one year⁴. While much of the research on scatter-hoarding was conducted on birds, this behavior is not specific to them⁴: squirrels⁵, chipmunks⁶, and even foxes^{7,8} stash food for times of need. Since caching sites cannot be defended, the success of this strategy often depends on an animal's ability to keep the stashes hidden and difficult to find⁵.

The ability to retrieve items from cache sites was shown to depend on spatial information such as visual cues. In Ref.⁹, black-capped chickadees (*Poecile atricapillus*) were placed in an enclosure, and their food-caching behavior was tracked. Rearranging objects around the enclosure greatly impaired the chickadees' ability to find their cache sites; manipulation of prominent global landmarks (large cardboard cutouts and a poster) had a much stronger effect on the birds' retrieval performance than small proximal objects (5 cm squares). Shifting objects by as little as 20 cm to the right significantly decreased the chickadees' ability to recover the food. Moreover, in almost 70% of the cases, the birds searched within 5 cm of the location implied by the more prominent landmarks, with a mean displacement of around 20 cm. Assuming this is approximately the caching resolution, the finding indicates that a small area of 10 × 10 m can hold as many as 2500 potential caching sites.

The hippocampus plays a central part in the remarkable cognitive feat of caching¹⁰. This is not surprising, given that the hippocampus is known to be involved in processing spatial information in the brain¹¹. A large subpopulation of neurons within the hippocampus in animals such as mice, rats, and bats exhibit *place-cell* behavior; that is, they increase in their spike rate in response to the animal's entering a specific region within a given site (usually, but not always, one region per cell). The region activating each place cell often changes when the animal moves to a new area, often in an unpredictable manner. However, if the animal returns to a previously visited site, the place cell's receptive fields revert to their previous arrangement, and this change happens almost instantaneously. This remapping of the receptive field within a given environment is mostly insensitive to landmark manipulations. In primates, we usually find a related type of cells referred to as spatial view cells.

¹Department of Cognitive and Brain Sciences, The Hebrew University of Jerusalem, Jerusalem, Israel. ²Department of Animal Sciences, The Hebrew University of Jerusalem, Rehovot, Israel. ✉email: oren.forkosh@mail.huji.ac.il

These cells respond remotely when an animal is gazing at a specific region, independently of the animal's location or head direction¹².

A well-known homolog to the mammalian hippocampus also exists in birds, with similar involvement in spatial and episodic memory¹³. The size of the hippocampus in birds was found to correlate with their ability to stash food. Although the interpretation of this correlation is under debate^{14,15}, animals that used more cache sites generally had a larger hippocampus than non-caching bird species^{16,17}. Even within the same species, the size of the hippocampus was found to be larger in individuals dwelling in harsher environments which makes them more dependent on cached food¹⁸. Moreover, hippocampal neurogenesis has a seasonal element and seems to correlate with caching activity throughout the year¹⁹. For many years, the spatially responsive cells found in avian brains were less related to a fixed position in space and more related to the challenge the animal faced such as the position of a goal within a maze²⁰. Only very recently the existence of place cells was demonstrated in the tufted titmouse (*Baeolophus bicolor*)²¹.

Taken together with the fact that the hippocampus is involved in memory, these observations have led researchers to hypothesize that caching requires some form of spatial and episodic memory²².

Yet as birds and other animals need an internal mechanism to guide them to stash food in specific locations, the same mechanism can also be used to direct them to the exact same locations while retrieving the food in that area. A mechanism, or mapping, that can facilitate efficient hiding and retrieval of multiple cache sites without relying on memory would need to have several basic properties. From a theoretical perspective, the class of methods that achieve this is known as hashing functions²³.

In this context, hashing refers to a class of functions that map arbitrarily complex data (such as images, texts, audio files, and others) to a fixed-size lower-dimensional representation. Computer applications often use hashes to store objects in memory efficiently by assigning them a unique identifier that maps directly to a specific memory location, effectively functioning as a retrieval shortcut. In the case of food caching, hashing can be used to map a set of landmarks within and around an area onto a selected caching site within that area (Fig. 1a). Efficient hash functions are designed to minimize the probability of different inputs being assigned the same output. This property reduces the likelihood of collisions and redundancies that can occur for animals, this allows them to utilize the entire area for caching while minimizing the probability of overlap and, consequently, reducing the risk of scrounging.

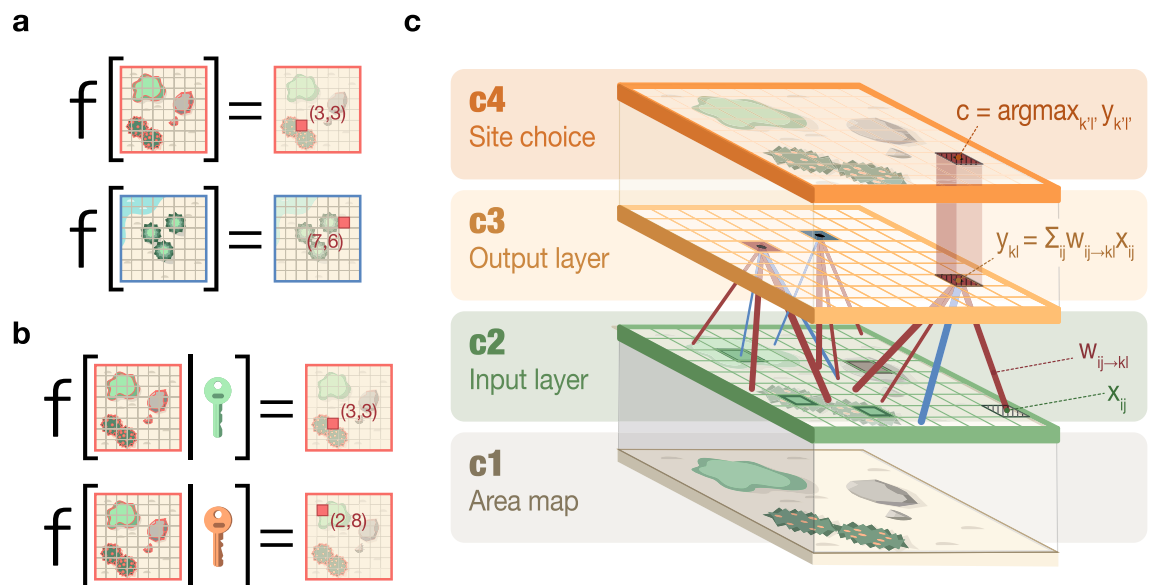


Figure 1. Neural Hashing Mechanism for Cache Site Selection. (a) The panel illustrates how a hash function can map complex, high-dimensional data, such as a detailed terrain map, to determine food cache locations. A red square highlights the cache site. (b) To ensure that the same data yields different mappings for different individuals, hash functions often use a unique 'private key'. However, it is possible to replicate this distinctiveness without explicitly incorporating a private key by using random parameters for each hash function. (c) A hash function can be replicated using a single-layer neural network. In this model, each input neuron, denoted by x_{ij} , corresponds to a specific grid cell overlaying the terrain map (panels c1 & c2). For illustrative purposes, we categorize terrain features into two types: 'plants' and 'rocks'. We assume an input neuron's sign indicates the type of terrain feature (positive for plants and negative for rocks) while its magnitude marks the prominence of the feature. The output neurons, represented by y_{kl} , are arranged in a grid that mirrors the input layer (panel c3). Each neuron in the input layer has a corresponding neuron in the output layer, with both representing the exact physical location on the map. These output neurons allocate a 'caching score' to every grid cell, and the cell receiving the highest score is selected as the cache site for food storage (panel c4). Sparse and randomly distributed synaptic connections ($w_{ij \rightarrow kl}$) bridge the input and output layers, facilitating this mapping process (panels c2, c3).

Hash functions often incorporate an additional factor—referred to as a private key—that renders the mapping unique to the key owner²³. Assuming no two individuals have the same key, it also means that the mapping will result in unique caching sites within the same area (Fig. 1b). Another valuable property of hash functions with private keys is that it is often difficult to infer the key from a small number of examples. So even if another animal uncovers one or more caches, it will not be able to deduce the location of all others.

Utilizing a pseudo-random hash function is less cognitively demanding than memorizing hundreds of thousands of stashing sites while still supporting existing empirical evidence. This hashing mechanism could function as a mnemonic aid²⁴ or might even negate the need for memory use. The decision-making process for selecting cache locations could rely entirely on identifying prominent, stable environmental features like trees and rocks, which serve as reliable reference points for later retrieval. Moreover, hash functions with multiple caching scores offer the added advantage of enabling animals to prioritize food based on spoilage risk or nutritional content. We suggest that hippocampal spatial cells are a likely mechanism capable of fulfilling this role. These cells, individualized in each animal, allocate scores and rankings to various locations, as reflected in their spiking rates. Persistent over time and adaptable to familiar terrains, these spatial cells' characteristics can provide the means for animals to locate their food caches effectively and discretely.

Results

Neural hashing function for cache site selection

A straightforward, parsimonious, and biologically plausible realization of hashing is through a neural network (Fig. 1c; for an in-depth description, please see the “Methods” section). In this construct, the input layer encodes key environmental landmarks (such as established vegetation, geological formations, and topographic features) that are consistent within a landscape (Fig. 2a,b). While the output layer designates the locations for food caches in that region (Fig. 2c). Both layers are arranged in a two-dimensional grid where each cell corresponds to a distinct location. For simplicity, we assume throughout this paper that the input and output layers have equal dimensions and each cell in the input layer has a corresponding cell in the output layer that points to the same location within the given area. The cache site is determined by the activity level of the output neurons, known as the cache score, with the site being allocated according to the neuron with the highest score.

In the model, a geographic area is divided into a grid, where each cell represents a specific coordinate and corresponds to an individual input neuron (Fig. 2a). These input neurons x_{ij} represent prominent and persistent environmental characteristics within each grid cell (Fig. 2b). The absolute value $|x_{ij}|$ indicates the prominence or saliency of the feature, while its sign indicates the type of feature. For ease of interpretation, plants are categorized as positive values and rocks as negative values. For example, a higher positive input value, such as $x_{ij} = 0.9$, would indicate a conspicuous tree at coordinates (i, j) , whereas a smaller positive value, such as 0.3, might indicate a small bush. Conversely, an input value of -0.7 would signify a large boulder. With additional input grid layers, other landmarks such as water sources or other geographical features can be incorporated into the model.

The output layer corresponds with the input layer's spatial grid and determines the location of the cache site, as shown in Fig. 2c. Cache sites are selected based on the coordinates indicated by neurons with the highest caching scores. Mathematically, the caching score of an output neuron y_{kl} at the coordinates (k, l) is the weighted sum of the input signals x_{ij} adjusted by the respective synaptic weight $w_{ij \rightarrow kl}$ yielding

$$y_{kl} = \sum_{i,j} w_{ij \rightarrow kl} x_{ij}.$$

The cache site location \bar{c} is determined by the output neuron with the highest score or

$$\bar{c} = \underset{(k,l)}{\operatorname{argmax}} y_{kl}.$$

For designating multiple cache sites within an area, the model can prioritize sites by the score of the neurons such that

$$\{y_{k_1 l_1} \geq \dots \geq y_{k_n l_n}, \text{ and } y_{k_n l_n} \geq y_{kl} \text{ for } \forall (k, l) \notin \{(k_1, l_1), \dots, (k_n, l_n)\}\}.$$

In this way, the network can rank potential cache sites and select the n most suitable locations based on their respective neural activation scores.

This network functions as a hashing mechanism, taking a landscape scattered with landmarks as its input. The number of landmarks it can process is limited only by the number of available cells in the input grid. This method of mapping items to indices is a common approach in neural networks and is often referred to as the ‘hashing trick’²⁵. The final output of the caching function is a two-dimensional vector corresponding to the coordinates of the designated cache site.

As we show later in the “Methods” section, the synaptic weights leading to each output neuron should ideally be orthogonal for the neural network to serve as an effective caching strategy. However, considering the limitations of biological systems, this requirement can be relaxed using an approximation. We can achieve a similar network behavior by randomly choosing synaptic weights that are uncorrelated, identically distributed, and have a zero average value. All our simulations are based on this approximation. Additionally, adding sparsity to the connections between the input and output layers can improve the performance of the network, as shown in Figs. 3 and 4.

Properties of caching networks

For a network to be effective as a caching mechanism, it must meet two critical criteria: first, it must minimize the chances of food caches being scrounged. Second, it must ensure the cache sites remain persistent despite

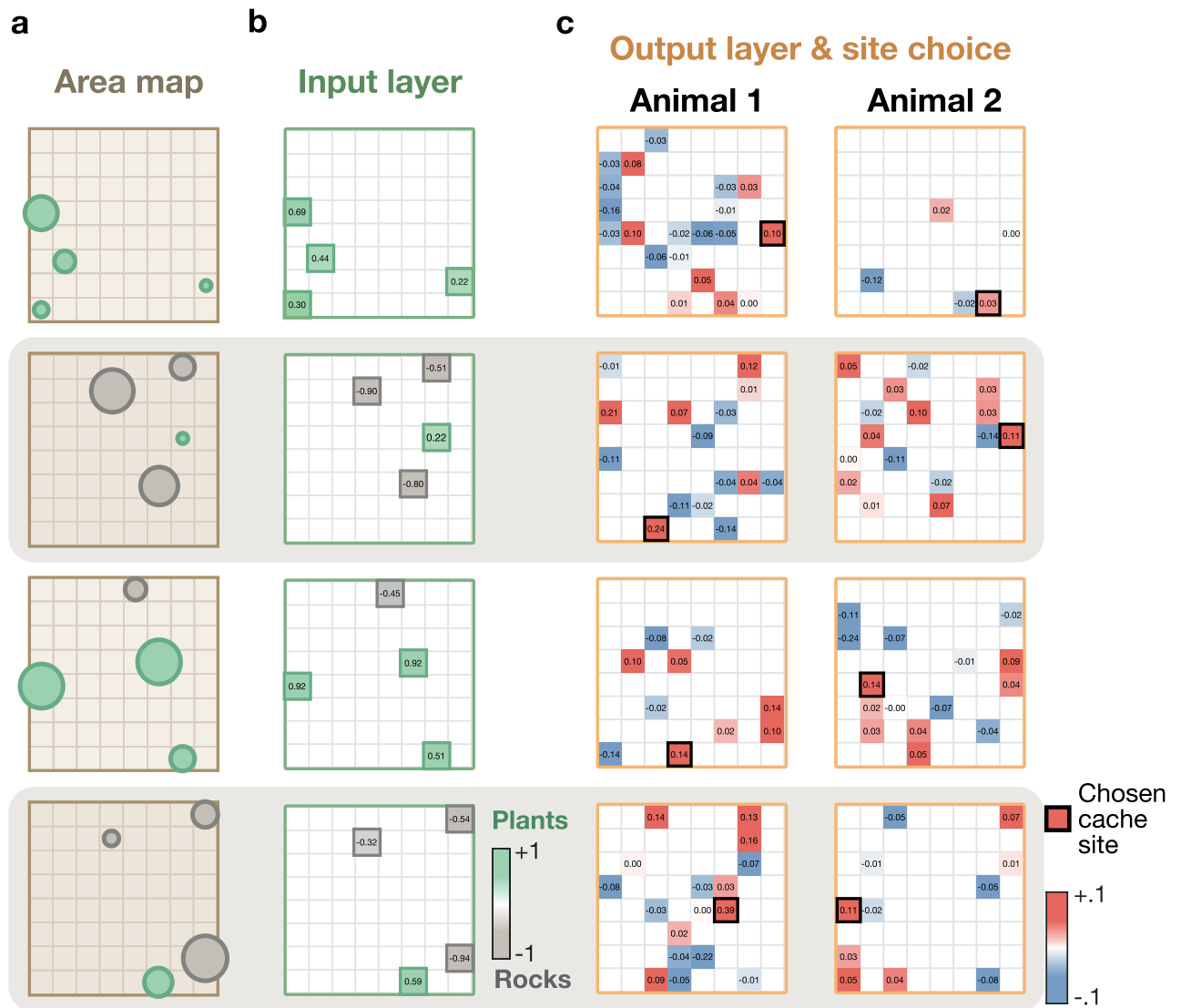


Figure 2. Caching network responses in different animal models. This figure illustrates the output of two distinct caching networks, referred to as ‘animal 1’ and ‘animal 2’, when subjected to various terrains. Each network employs a 8×8 grids for both the input and output layers. Synaptic connections between these layers are established so that each output neuron is connected to four randomly selected input neurons from the 64 possible synaptic connections. The weights for these active synapses are derived from a uniform distribution (see “Methods”). Panels (a) and (b) present the terrain inputs along with the neural signals processed by the caching networks. In our simulations, four cells were randomly chosen to represent prominent spatial features, with values assigned from a uniform distribution ranging from -1 to 1 . Positive values, representing ‘plants’, are depicted in green, while negative values, representing ‘rocks’, are shown in gray. Panel (a) illustrates the visual configuration of the terrain, whereas panel (b) displays the activation patterns in the input layer. The color and intensity of each cell in panel (b) indicate the type and strength of the input, respectively. Panel (c) visualizes the outputs of each animal’s network, showing different caching scores determined by the network’s unique, randomly assigned synaptic weights. In this panel, red denotes positive output values and higher caching scores, blue represents negative lower caching scores, and white signifies values around zero. The cell with the highest score is highlighted with a bold border, indicating the preferred cache site.

expected environmental changes; otherwise animals will be unable to retrieve their food. We explore these properties empirically in the following sections and analytically in the “Methods” section.

Food cache safety

Caching networks must be unique, akin to private keys, so that each animal’s cache sites remain private. In addition, the distribution of potential cache sites generated by different networks within a given area should be nearly uniform, mitigating biases or redundancies that might lead other animals to discover these sites and reducing the chance of overlapping cache sites.

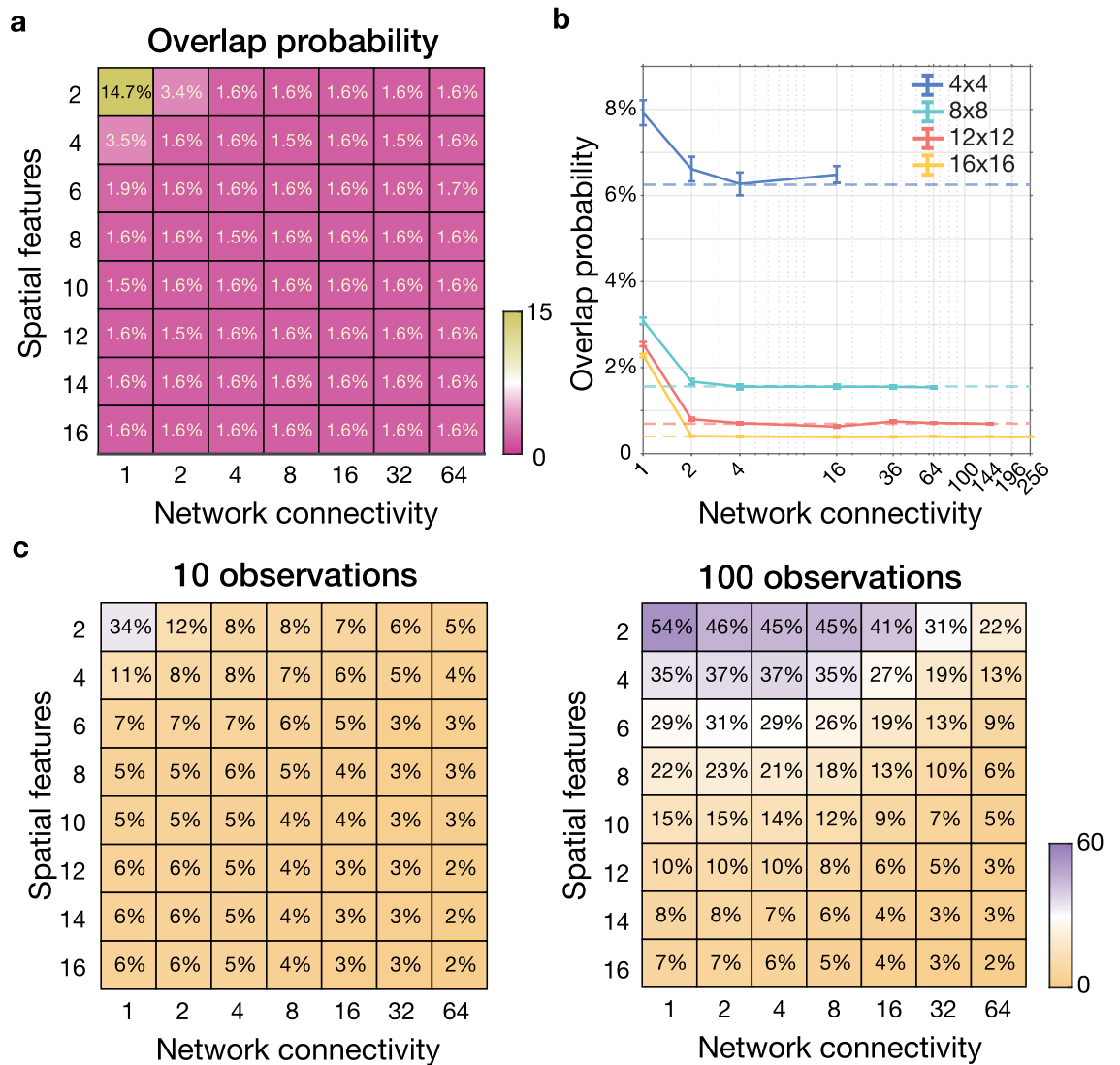


Figure 3. Cache Site Uniqueness and Security from Potential Scroungers. **(a, b)** Uniqueness of Cache Selection. Panel **(a)** illustrates the probability of overlapping cache sites between two distinct caching networks. Both networks were defined on an 8×8 grid, and varied in connectivity and the number of environmental features. Overlap probabilities were calculated by simulating 200 random network pairs, each evaluated on 100 random terrains, with a baseline overlap chance of approximately 1.56% (refer to the “Methods” section for additional details). The color gradient represents overlap probabilities on the graph, ranging from 0 (magenta) to 15% (green). Panel **(b)** illustrates overlap probabilities between networks of varying grid size: 4×4 (blue), 8×8 (cyan), 12×12 (red), and 16×16 (yellow). The error bars indicate the standard error for each data point. The dashed horizontal lines, color-coded to match each network size, mark the expected chance level of overlap for the respective grid sizes. These values were derived from testing 200 network pairs against 1000 random terrains. **(c)** Irreversibility of the caching function. The panel evaluates the irreversibility of the caching function by measuring the probability of inferring another animal’s cache sites from a limited set of examples. To infer the cache sites, a support vector machine (SVM) with a Gaussian kernel was used. The percentages indicate the probability that the SVM was correct in its site location prediction. We trained the algorithm on 10 examples (left graph) and 100 examples (right graph). The probabilities were determined as the average of 200 simulated networks. As before, the network was set to an 8×8 grid, with a baseline chance level of approximately 1.56%. Prediction success is visualized using a color spectrum from 0 (yellow) to 60% (purple).

To determine the probability of two independent caching networks choosing identical cache sites within a specific terrain, we conducted a series of simulations (Fig. 3a,b). Simulations were performed using caching networks with grid sizes of 8×8 (Fig. 3a); however, we also explored the effect of grid size choice (Fig. 3b). In these tests, we manipulated the number of synaptic connections to each neuron in the output layer, which we refer to as network connectivity. The connectivity ranged from a minimum of one connection to a maximum of 64 connections, the latter representing a fully connected network for an 8×8 grid. In Fig. 3a, the simulations also varied in the number of random features generated in each terrain, from 2 to 16, to observe how these variations affected network cache site choice. The number of spatial features in Fig. 3b was set to 4.

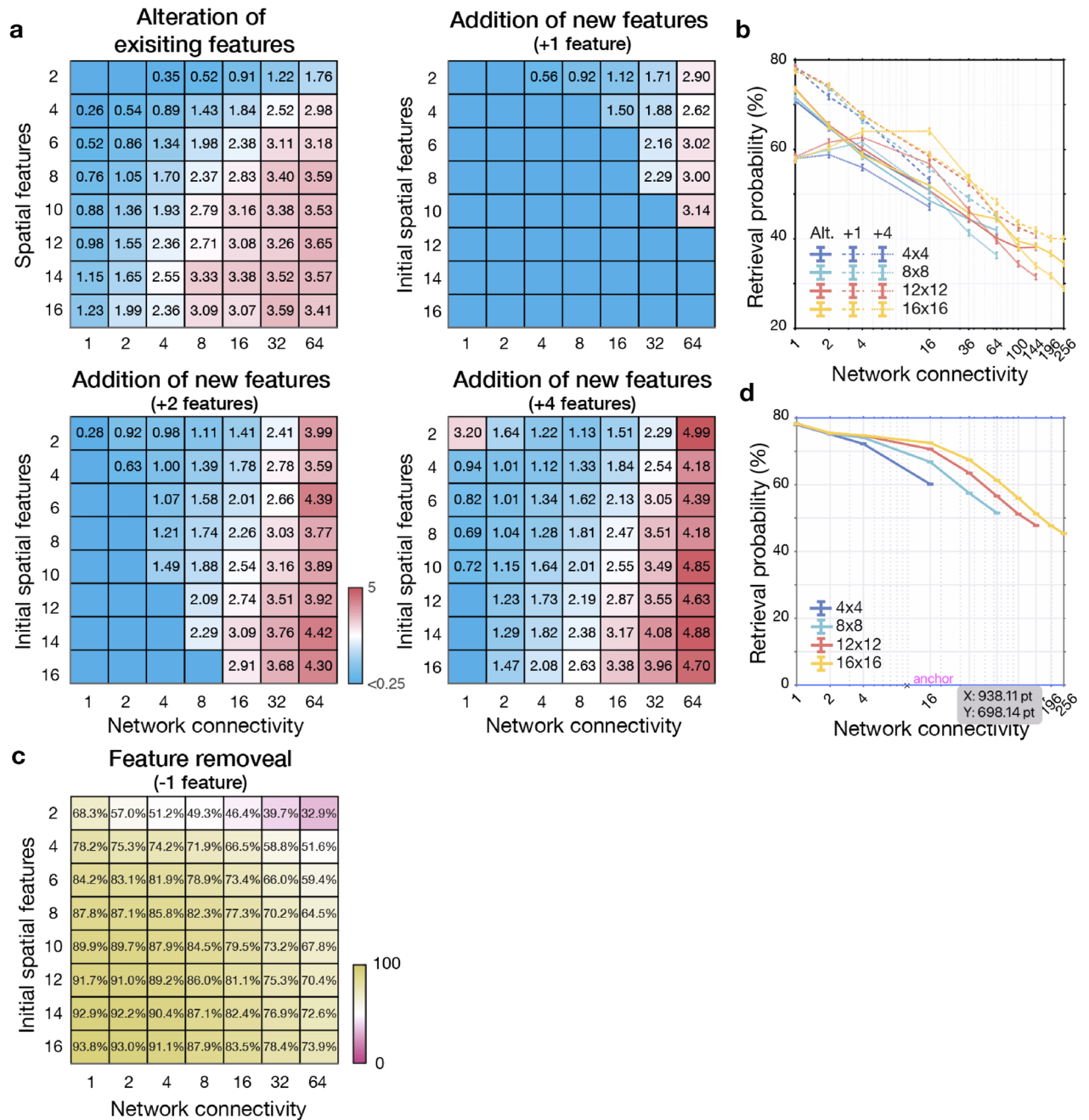


Figure 4. Caching stability and resilience. (a) Caching stability and resilience to environmental changes were determined as the minimal signal-to-noise ratio (SNR) that is needed to maintain a cache retrieval probability of 50%. A lower SNR indicates that the network is more stable even with higher noise levels. In this panel, we simulated two classes of environmental changes: the alteration of existing spatial features (top left graph) and the addition of 1, 2, or 4 new features (other graphs, as described in each subtitle). The colorbar maps out the range of minimal SNRs necessary, from values of 0.25 or lower (blue) to 5 (red), across 5000 simulation trials per data point. SNRs less than 0.25 were not annotated. (b) The probability of cache retrieval in networks of different sizes and connectivity (as in Fig. 3b), under two types of environmental changes: alteration (solid lines) and addition of 1 or 4 features (striped and dotted lines respectively). Each data point was determined from 5000 simulations. The error bars represent the standard error. Panels (c) and (d) illustrate the cache retrieval probability after removing a spatial feature. In panel (c), an 8 × 8 network was used to test different initial numbers of spatial features and connectivity levels. The color gradient in panel (c) charts retrieval probabilities from 0 (magenta) to 100% (green). Panel (d) compare networks of different sizes on random terrain with four initial spatial features (as in Fig. 3b and panel (b)).

Significant overlap in cache site selection were rare unless both the network's connectivity was very low (4 connections or fewer) and the terrain was relatively featureless (6 features or fewer). The probability of overlap hovered around chance level in most cases, which is approximately 1.56% for an 8×8 grid (see "Methods"). As the chance level sets a lower bound for coincidental overlap, we found that larger networks would produce a lower likelihood of cache site overlap (Fig. 3b) as could be anticipated.

Another important aspect of caching networks is that their complexity should be sufficient to prevent adversaries from inferring an animal's cache locations. To investigate this, we tested the ability to predict a cache site from a limited set of observations (Fig. 3c). We used a Support Vector Machine (SVM) with a Gaussian kernel (see "Methods"), an efficient and widely used nonlinear method^{26,27}. In contrast to alternatives (such as neural networks), this method does not rely on parameter tweaks that might bias the results.

Figure 3c shows that simplifying the network by reducing synaptic connections between layers made it easier to 'reverse-engineer' cache locations, making sparse networks more predictable. In contrast, landscapes with a higher number of unique landmarks posed a greater challenge for the predictive algorithm, suggesting that animals should favor these types of areas when caching.

Caching stability and resilience

Caching networks require a unique level of resilience against environmental changes. Unlike hash functions used in computing, which are designed to yield significantly different results from minimal input changes²³, food caching networks require a design focused on stability. These networks should be impervious to environmental changes, such as seasonal growth or decay of vegetation, the emergence of new environmental features, as well as other variations. By maintaining this steadfastness, the cognitive map used to locate caches remains constant and reliable, allowing animals to reliably retrieve their stored food regardless of the ever-changing world.

We evaluated the network's resilience to three kinds of environmental changes: alteration of existing spatial features (Fig. 4a,b), addition of new features (Fig. 4a,b), and removal of existing elements (Fig. 4c,d). The effects of these modifications and additions were determined by the minimum signal-to-noise ratio (SNR) that resulted in a recall probability below 50% (Fig. 4a,b; see the "Methods" section for details). Generally, an SNR of one means that the noise introduced into the model is equal to the original signal's amplitude, while lower SNR values indicate that the network can withstand noise greater than the signal. In accordance with previous experiments, these tests employed an 8×8 grid as shown in Fig. 4a, while Fig. 4b compares retrieval performance across networks with varying grid sizes. Sparser networks applied to less complex terrains were more resilient and could withstand lower SNRs (Fig. 4a). However, when the number of new features introduced into the terrain exceeded the original count, sparse networks and simple terrains showed decreased robustness. Network size did not affect retrieval success rate except when the number of new features was high (+4), in which case larger networks were more effective.

Sparsier networks also performed better on terrains in which features were removed (Fig. 4c). As the complexity of the terrain increased, retrieval performance also improved. In addition, larger networks were more resilient to feature removal (Fig. 4d).

Together with our previous findings, this new insight highlights the trade-off between a network's resilience to environmental changes and its ability to safeguard an animal's caches. This suggests that variations in animal behavior and habitat may influence the architecture of these networks, tailoring them to balance the demands of stability against the risk of scrounging.

Discussion

Our study suggests that hippocampal spatial cells serve as the brain's equivalent of hash functions, enabling animals to effectively hide food and retrieve it later without requiring any form of long-term memory. It is proposed that place cells guide animals towards potential cache sites when they arrive at a specific area, and not just point to the animal's position. In addition, recent studies have revealed that some hippocampal cells respond to inanimate objects in a way that is consistent with the way our model's input layer encodes salient features²⁸. These findings imply that we might find the equivalence of both the input and output layers of our model in the hippocampus.

While we often find sequential activation of place cells along trajectories, our model does not necessarily conflict with this view¹¹. Instead, it could complement it by suggesting that while animals traverse their environment, they sequentially activate a series of place cells that, in turn, contribute to the building of a cognitive caching map over time. The caching function in this context does not rely on an instantaneous, all-at-once snapshot of space, but rather on a cumulative process that integrates these linear trajectories over multiple exposures to the environment. It is also plausible to consider that certain place cells fire in a directional manner, similar to how a navigation application operates. Rather than encoding the animal's location, these cells might provide relational cues, guiding the animal toward a target destination based on its current position.

The model we presented is a simplified abstraction rather than a precise depiction of the neural circuitry within the brain. It serves as a foundational or "lower bound" framework, outlining the minimal architecture required for functional caching. We have demonstrated that even a sparsely connected network consisting solely of input and output layers can function as a caching mechanism. The reality of the brain's neural circuits is likely to be more complex, possibly encompassing a greater array of features, such as additional types of features, or encompassing mechanisms for translating reference spaces.

The hashing network we constructed closely aligns with a mathematical concept known as compressed sensing. Compressed sensing is a signal processing technique (rather than a data storage and retrieval method like hashing) that allows for the reconstruction of sparse signals from a small number of random projections²⁹. This technique uses the sparsity of the signal in a specific basis, enabling efficient data acquisition and reconstruction. The core idea is that, with high probability, random projections of a sparse signal retain sufficient information

for accurate reconstruction, provided these projections are sufficiently incoherent with the sparsity basis. In the context of our model, the use random projections in the hashing function ensures that each environmental feature is mapped uniquely and efficiently, much like how compressed sensing reconstructs a signal from limited data. By assuming orthogonality between the projections, our model can uniformly distribute cache sites across the environment, thereby minimizing the likelihood of collisions and ensuring that the retrieval process is robust to environmental changes. Several previous studies have suggested that compressed sensing is a mechanism utilized by the brain for dimensionality reduction²⁹ including in the hippocampus³⁰, for learning the statistical properties of signals³¹, and for short term memory³².

Caching networks can be tweaked by changing their sizes and connectivity, and the results suggest that a single, uniform configuration may not be suitable for all. There appears to be a delicate balance between enhancing the network's security and improving its robustness against environmental variability. These conclusions imply that animals from distinct habitats may exhibit variations in their neural network architectures, tailored to their specific environmental challenges and needs.

Greater efficiency of caching can be achieved by distributing multiple items across the terrain, utilizing a selection of output neurons that show high levels of activity. This tactic can also allow animals to prioritize the caching of food items based on their nutritional value or perishability—for instance, placing highly perishable items like insects, rather than more durable items like seeds, at locations indicated by neurons with higher caching scores. This method improves the chances of these items being retrieved and being retrieved first. To avoid revisiting previously excavated cache sites, animals need only remember the score of the last site dug up, and then choose new sites with a lower score, effectively using the scores as a system of bookmarking for cache locations.

Animals might implement additional strategies for improving the functionality of caching networks. For instance, to enhance the network's resilience, animals might prefer terrains where there is a noticeable discrepancy in activation between the highest-scoring output neuron and the subsequent, lower-scoring ones. Such a strategy would help to ensure that environmental alterations do not affect the predetermined location of the cache site.

Whether a caching map is allocentric, relating to external coordinates, or egocentric, centered on the animal, it must remain unchanged regardless of the animal's position. An effective method to ensure this invariance, as detailed in a notable study on geometric hashing³³, involves selecting two distinctive landmarks within the environment. These landmarks are then utilized as reference points to scale and orient all other landmarks in the terrain. The directional axis is defined by the vector joining these two landmarks, and the distance between them establishes the unit of scale. Adopting this method results in a model that is insensitive to all affine transformations.

While our study focuses on scattered hoarding animals, the underlying mechanism we propose could facilitate species-specific navigation within complex environments in other species. It is possible that simple and consistent hash functions shared among conspecifics can play an important role in guiding individuals to common meeting places, whether for mating purposes or other social interactions. This functional, as opposed to memory-dependent, mechanism might also naturally direct migratory species to their destinations, implying an inherent capacity for navigation.

Furthermore, considering the hippocampus's role not just in spatial orientation but also in the processing of abstract knowledge¹¹, we can speculate that the decision-making process in animals—and potentially humans—may be influenced by similar neural pathways. Given that diversity is a hallmark of living systems, the unique individualistic behaviors observed in humans may share roots with these neural strategies, suggesting an interesting intersection of biology and behavior.

While this study is conceptual, it does lead to several testable predictions. Firstly, by understanding the firing patterns of spatial cells, we might be able to identify the locations of cache sites. Additionally, with a thorough knowledge of the remapping process between different locations, this information could be used to predict the location of cache sites within new areas that an animal has not yet explored. And lastly, we anticipate that cache site locations within a specific area will remain stable over successive caching and retrieval cycles.

Methods

Memoryless network-based caching

We define a caching mechanism using a single-layer linear neural network followed by a non-linear decision rule that determines the location of the cache sites. In this model, the *input layer* forms a two-dimensional grid, with each cell representing a specific location in the animal's current environment and encoding salient geographical features. Locations with prominent landmarks are allocated either positive or negative values based on the feature type and its relative prominence. The output layer mirrors the spatial organization of the input layer, so that each cell in the input layer is paired with a cell in the output layer, although the two cells are not necessarily connected.

The linear part of the network is used to assign a 'caching score' to each cell in the current environment. The outputs of the network are calculated as follows

$$y_{kl} = \sum_{ij} w_{ij \rightarrow kl} x_{ij}, \quad (1)$$

where x_{ij} is the value of the input cell at location (i, j) and y_{kl} is the output neuron with coordinates (k, l) . We denote by $w_{ij \rightarrow kl}$ the weight of the connection from the input neuron x_{ij} to the output neuron y_{kl} . To determine the location of the cache site \bar{c} , the network employs a non-linear activation function, namely the "argmax" function

$$\bar{c} = \operatorname{argmax}_{(k,l)} y_{k,l}. \quad (2)$$

This function sets the target cache site as the coordinates that correspond to the neuron with the highest caching score within the grid.

In this paper, we make the following assumptions regarding the network, which we discuss in more detail in subsequent sections. First, we assume the inputs to the network are independent and identically distributed, which greatly simplifies our analysis. Second, we show that, ideally, the synaptic weights leading to each output neuron should be orthogonal. However, we can approximate this property by randomly selecting weights that are uncorrelated, identically distributed, and centered around a zero mean. Finally, we assume the network is sparse, meaning only a small subset of the input neurons is connected to each output neuron. This step reduces the complexity of the network while improving some aspects of its performance. We will show that these constraints make the mapping efficient according to the criteria presented in the following section.

Caching efficiency

For the network to function as the basis of an effective caching strategy, it must satisfy two essential conditions, which we explore theoretically and through simulations:

- (1) **Safety and Uniqueness:** The network must ensure that the locations of caches are not discovered and exploited by competitors. To achieve this:
 - a. **Distinct Outputs:** The network's outputs must be distinct for each area and unique for each individual animal in order to prevent others from finding the cache sites.
 - b. **Complex Design:** The network's design must be complex enough to safeguard food caches from being exposed by other animals. Observing the animal's behavior or randomly searching for caches should not endanger other caches by allowing adversaries to decipher the network's workings through 'reverse engineering'.
- (2) **Stability and Resilience:** Animals need to locate their cache sites reliably, even when the environment is constantly changing. To achieve this, the network must be resilient to variations in its inputs, which reflect the natural fluctuations of the environment.

Properties of the caching network

Choice of synaptic weights

To simplify the analysis and without loss of generality, we introduce a one-dimensional version of the network described in Eqs. (1) and (2). In this simpler model, both inputs and outputs are represented as vectors with n elements instead of being structured in a two-dimensional grid. Equation (1) reduces to

$$y_k = \sum_i w_{ik} x_i, \quad (3)$$

where $\bar{x} = (x_1, \dots, x_n)$ and $\bar{y} = (y_1, \dots, y_n)$ represent the input and output vectors, respectively, with w_{ik} denoting the weight of the connection from input neuron x_i to output neuron y_k . In addition, following Eq. (2), we apply a non-linear function to the output layer, determining the cache site by selecting the index of the output neuron with the highest score

$$c = \operatorname{argmax}_{k'} y_{k'}. \quad (4)$$

Here, c is the index of the chosen cache site.

To mitigate the risk of exposing cache sites, it is vital to limit the information available to other animals about these locations. This involves maximizing the entropy of cache site distribution, essentially ensuring an even spread of sites. To achieve this objective, we configure the network's connections to be random, independent, and identically distributed (IID) with a zero mean, as explained further below. This random network setup minimizes the chance of multiple animals converging on the same cache site.

In mathematical terms, assuming that the inputs are independent and identically distributed (IID) with a zero mean implies that any two input neurons, denoted as x_i and x_j , have a covariance of zero. In other words, for all $i \neq j$ the covariance $\operatorname{cov}(x_i, x_j)$ is given by

$$\operatorname{cov}(x_i, x_j) = \mathbb{E}[x_i \cdot x_j] - \mathbb{E}[x_i] \mathbb{E}[x_j] = 0.$$

Here, \mathbb{E} denotes the expected value.

To ensure a uniform distribution of cache sites, it's important that the output neurons are also independent and equally distributed. In such a scenario, the symmetry among neurons leads to a uniform distribution of outcomes according to Eq. (2). The covariance between any two output neurons, denoted as y_k and y_l can be expressed using the values of input neurons and synaptic weights as follows

$$\begin{aligned}
\text{cov}(y_k, y_l) &= \mathbb{E}[y_k \cdot y_l] - \mathbb{E}[y_k]\mathbb{E}[y_l] = \mathbb{E}\left[\sum_i W_{i \rightarrow k} x_i \cdot \sum_j W_{j \rightarrow l} x_j\right] - \mathbb{E}\left[\sum_i W_{i \rightarrow k} x_i\right]\mathbb{E}\left[\sum_j W_{j \rightarrow l} x_j\right] \\
&= \sum_{i,j} W_{i \rightarrow k} W_{j \rightarrow l} \mathbb{E}[x_i \cdot x_j] - \sum_{i,j} W_{i \rightarrow k} W_{j \rightarrow l} \mathbb{E}[x_i]\mathbb{E}[x_j] = \sum_{i,j} W_{i \rightarrow k} W_{j \rightarrow l} (\mathbb{E}[x_i \cdot x_j] - \mathbb{E}[x_i]\mathbb{E}[x_j]) \\
&= \sum_i W_{i \rightarrow k} W_{i \rightarrow l} (\mathbb{E}[x_i^2] - \mathbb{E}^2[x_i]) = \sigma_x^2 \sum_i W_{i \rightarrow k} W_{i \rightarrow l}
\end{aligned}$$

where σ_x^2 represents the variance of the input neurons, which is consistent across all inputs since they are identically distributed by definition.

This result indicates that the rows of the weight matrix W must be orthogonal, meaning each row is perpendicular to the other, for the output neurons' activities to be uncorrelated. However, for the distribution of the cache sites to be uniform across the network, it is also necessary that these weights have the same variance. In other words, they should form an orthogonal basis with a uniform norm. Mathematically, this requirement can be expressed for any output neurons y_k and y_l as

$$\text{cov}(y_k, y_l) = \sigma_x^2 \sum_i W_{i \rightarrow k} W_{i \rightarrow l} = \begin{cases} 0 & \text{if } k \neq l \\ n\sigma_x^2\sigma_w^2 & \text{otherwise} \end{cases} \quad (5)$$

where σ_x^2 is the variance of the input neurons, σ_w^2 is the variance of the synaptic weights, and n is the dimension of the network.

Since orthogonality is less likely to emerge in a biological network, we approximate this condition by randomly and independently setting the synaptic weights. These weights are distributed around a mean of zero and a variance of σ_w^2 . This approximation becomes increasingly precise as the network dimension n grows while maintaining the same covariance level. We can express this approximation analytically when the weights are sampled from a normal distribution. In this scenario, the variance of the sample variance is calculated as $2\sigma_w^4/(n-1)$, which diminishes towards zero as n increases.

Network sparsity

In addition to randomly selecting synaptic weights, we also randomly prune connections between the layers to create a sparse network. This pruning process simplifies the network and, as the simulations demonstrate, can improve performance. We measure the network's connectivity by the average number of input neurons connected to each output neuron. The pruning method we used is explained in detail in the next section.

Pruning affects several aspects of the network's behavior. On one hand, it simplifies the network's structure, but this simplicity might come at a cost, as it could increase vulnerability to reverse engineering attempts. Furthermore, excessive pruning can result in input neurons being disconnected from any output neuron, creating "blind spots" that prevent the model from responding to certain spatial cues.

Assuming a neuron's probability of connection is p , the likelihood of an input not being linked to any output is $(1-p)^n$ for a network of size $n \times n$. Assuming the number of disconnected inputs follows a Binomial distribution, the mean number of disconnected inputs is $n^2(1-p)^n$. For example, consider a network with dimensions of $8 \times 8 = 64$ where each output neuron has 4 random inputs, so $p = 4/64$, the mean number of non-connected inputs is approximately $1 \cdot (64 \cdot (1 - \frac{4}{64})^{64}) \approx 1.03$.

Terrain and network simulations

To empirically evaluate the properties of caching networks, we conducted a series of simulations. In these simulations, the environment was represented by an $n \times n$ grid, which depicted the spatial layout of the area. Within this grid, we randomly selected a subset of cells to represent prominent features in the area. The characteristics of these features, including their prominence and sign, were assigned values drawn from a uniform distribution $\mathcal{U}(-1, 1)$ covering a range from -1 to 1 . The number of features for each scenario is explicitly stated in the figures, with a default value of four features set for illustrative purposes.

The neural networks were constructed by randomly establishing synaptic connections between input and output neurons. Assuming both layers have similar dimensions, a grid of size $n \times n$ resulted in a total of n^4 possible connections, with weights randomly sampled from a uniform distribution $\mathcal{U}(-1, 1)$, spanning values between -1 and 1 . After this initial setup, the synaptic weights to each output neuron were randomly pruned, ensuring each output neuron had a consistent number of incoming connections. We refer to the number of incoming connections as the network's connectivity. For instance, a fully connected network has a connectivity of n^2 . By default, unless stated otherwise, the grid size was set to 8×8 and the connectivity level was set to 4.

These simulations were conducted to provide insights into the robustness and efficiency of neural networks associated with caching behavior in animals. The caching mechanism was examined under various environmental and structural conditions by varying the number of features and the connectivity level. The following sections detail the different tests.

Overlap probability of individual networks

An important requirement of caching is that the probability of two individuals choosing the same site is close to chance, thus minimizing the probability of resource exploitation or overlap in cache sites. To demonstrate this property and its dependency on the network and landscape properties, we ran a series of simulations (Fig. 3a,b).

In each iteration, we created two networks with distinct keys by randomizing the selection of synapses and their corresponding weights. We then calculated the proportion of instances where both networks selected the same cache site. This process allowed us to assess the effectiveness of our network design in maintaining the security of cache sites against potential predictability. Using a $n \times n$ grid means that the chance level, or the probability of randomly choosing the same site, is equal to $1/n^2$.

In Fig. 3a, we simulated 200 pairs of random 8×8 networks, and tested each pair on 100 randomly generated terrains. The baseline probability, or chance level, for overlap in this setup was about $1/8^2 = 1/64 = 1.56\%$. Figure 3b expands on this by examining networks of varying sizes, specifically 4×4 , 8×8 , 12×12 , and 16×16 grids. The corresponding chance levels for these networks were approximately 6.25%, 1.56%, 0.69%, and 0.39%, respectively. For each grid size, we ran simulations on 200 pairs of networks across 1000 randomly generated terrains. The increase in network size necessitated a higher number of simulations to obtain statistically significant results.

Irreversibility of the caching function

Irreversibility in caching functions ensures that deducing the output of the function based on a limited number of observations is difficult. This feature is essential for protecting cached resources from being discovered and exploited by others.

To evaluate the irreversibility of our caching function, we used regression analysis to predict cache site locations. This process, akin to "reverse engineering," aims to reconstruct the caching function using a finite set of examples. For this task, we used support vector machines (SVMs) with a Gaussian kernel²⁷. Support Vector Machines (SVMs) are a type of supervised machine learning algorithm used for classification and regression tasks. They work by finding the hyperplane that best separates different classes in the feature space, maximizing the margin between the closest points of the classes, known as support vectors. SVMs are effective in high-dimensional spaces and for cases where the number of dimensions exceeds the number of samples.

While a neural network could have been used instead of the SVM, given the limited number of examples SVM provided better performance (not shown) and depends on fewer parameters, reducing the risk of over-fitting. For the SVM, the sole parameter was the box constraint C , which determines the penalty for misclassifications. Here, we set C to $iqr(x)/1.349$ where iqr denotes the interquartile range or the spread of samples between the 25th and 75th percentiles. This is a common choice for the box constraint since it is equal to the standard deviation for a normal distribution ($\sigma = iqr(x)/1.349$).

The effectiveness of the regression was determined by its ability to predict the cache site selected by the original network. The SVM was trained using either 10 or 100 examples of terrain and cache site pairs and then tested on 100 novel terrains. A total of 200 simulations were conducted for each variant of network connectivity and landmark density, with a network size of 8×8 .

Caching stability and resilience

In computer science applications, hash functions are typically designed to be highly sensitive to input changes, which is crucial for ensuring security and data integrity. A hash function used for food caching, on the other hand, must be highly resilient to small variations in its inputs. This ensures that the output remains consistent despite natural environmental changes, such as seasonal variations or the appearance of new landmarks. To evaluate this resilience, we simulated environmental changes in three ways:

1. *Changes to existing features.* This simulates alterations to existing environmental features, such as the growth of trees. Variability was introduced by adding a uniformly distributed noise $n_{ij} \sim \mathcal{U}(-s, +s)$, to each non-zero valued feature x_{ij} , resulting in a new input value $x_{ij} + n_{ij}$. Here, s signifies the maximum amplitude of the noise.
2. *Emergence of new features.* This simulates events where some elements only appear during recall. For this simulation, we randomly selected unoccupied cells (where $x_{ij} = 0$) and introduced new landmarks by sampling values from a uniform distribution $n_{ij} \sim \mathcal{U}(-s, +s)$, and setting $x_{ij} = n_{ij}$. As before, s is the maximum amplitude of the noise.
3. *Disappearance of features.* This simulates events where one or more environmental features disappeared. Here, randomly selected occupied cells (where $x_{ij} \neq 0$) are changed to zero ($x_{ij} = 0$).

The impact of the additive noise under scenarios 1 and 2 was quantified by determining the minimum signal-to-noise ratio (SNR) necessary for successful food retrieval. We set a threshold of 50% retrieval success, defining success as the network's ability to select the same cache site after the introduction of noise as it did initially. The maximum amplitude of the noise (s) was determined empirically through the simulations; the values of s ranged from 0.05 to 2 in increments of 0.05 (0.05, 0.10, 0.15, ..., 2), and the highest value that met the 50% retrieval rate was chosen as the noise amplitude and used to determine the SNR.

The SNR was calculated using the following formula

$$SNR = \frac{\text{var}\left(\sum_{i,j} |x_{i,j}|\right)}{\text{var}\left(\sum_{i,j} |n_{i,j}|\right)},$$

where $\text{var}()$ represents the sample variance computed over multiple simulations. Assuming for the k 'th simulation the total signal amplitude is given by $S_k = \sum_{i,j} |x_{i,j}|$, the total variance is calculated as $\text{var}(S) = \frac{1}{K-1} \sum_{k=1}^K (S_k - \bar{S})^2$, where K is the total number of simulations and \bar{S} is the sample mean of the signal levels across all simulations. Similarly, the variance of the noise levels is computed for the mean noise level N_k which is defined as $N_k = \sum_{i,j} |n_{i,j}|$.

An SNR of 0.5, for example, indicates that the signal strength can be, on average, half of the noise level and still achieve a 50% retrieval success rate. Generally, a lower SNR signifies a more resilient system that can handle noisier environments.

We carried out 5000 simulations for each combination of network connectivity, number of features, and level of noise to determine the point at which the retrieval success rate dropped below the 50% threshold (Fig. 4a).

In addition, we tested the effect of network size by measuring recall probability when the SNR was set to 2 (Fig. 4b). As with previous tests, each data point on the graph was determined from 5000 simulations.

Scenario 3 was tested by randomly setting a non-zero input cell to zero and measuring the percentage of cache sites that did not move after the change. We generated 200 simulated networks and tested each one on 1000 terrains with a single feature omitted (Fig. 4c,d).

Data availability

All data, code, and simulations will be available upon request. Please contact Oren Forkosh at oren.forkosh@mail.huji.ac.il for code and data.

Received: 20 August 2023; Accepted: 17 July 2024

Published online: 06 August 2024

References

- Stevens, M. & Merilaita, S. Animal camouflage: Current issues and new perspectives. *Philos. Trans. Royal Society B Biol. Sci.* **364**, 423–427 (2009).
- Ruxton, G. D., Allen, W. L., Sherratt, T. N. & Speed, M. P. *Avoiding Attack: The Evolutionary Ecology of Crypsis, Aposematism, and Mimicry*. (Oxford University Press, 2019).
- Butler, R. K. & Finn, D. P. Stress-induced analgesia. *Prog. Neurobiol.* **88**, 184–202 (2009).
- Smith, C. C. & Reichman, O. J. The evolution of food caching by birds and mammals. *Annu. Rev. Ecol. Syst.* **15**, 329–351 (1984).
- Jacobs, L. F. & Liman, E. R. Grey squirrels remember the locations of buried nuts. *Anim. Behav.* **41**, 103–110 (1991).
- Yahner, R. H. The adaptive nature of the social system and behavior in the Eastern Chipmunk, *Tamias striatus*. *Behav. Ecol. Sociobiol.* **3**, 397–427 (1978).
- Tinbergen, N. Von den Vorratskammern des Rotfuchses: (*Vulpes vulpes* L.). *Zeitschrift für Tierpsychologie* **22**, 119–149 (1965).
- MacDonald, D. W. Food caching by red foxes and some other carnivores. *Z. Tierpsychol.* **42**, 170–185 (1976).
- Herz, R. S., Zanette, L. & Sherry, D. F. Spatial cues for cache retrieval by black-capped chickadees. *Animal Behav.* **48**, 343–351 (1994).
- Brodin, A. The history of scatter hoarding studies. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **365**, 869–881 (2010).
- Behrens, T. E. J. *et al.* What is a cognitive map? Organizing knowledge for flexible behavior. *Neuron* **100**, 490–509 (2018).
- Rolls, E. T. Spatial view cells and the representation of place in the primate hippocampus. *Hippocampus* **9**, 467–480 (1999).
- Jarvis, E. D. *et al.* Avian brains and a new understanding of vertebrate brain evolution. *Nat. Rev. Neurosci.* **6**, 151–159 (2005).
- Roth, T. C., Brodin, A., Smulders, T. V., LaDage, L. D. & Pravosudov, V. V. Is bigger always better? A critical appraisal of the use of volumetric analysis in the study of the hippocampus. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **365**, 915–931 (2010).
- Sherry, D. F. Neuroecology. *Annu. Rev. Psychol.* **57**, 167–197 (2006).
- Krebs, J. R., Sherry, D. F., Healy, S. D., Perry, V. H. & Vaccarino, A. L. Hippocampal specialization of food-storing birds. *Proc. Natl. Acad. Sci. USA* **86**, 1388–1392 (1989).
- Sherry, D. F., Vaccarino, A. L., Buckenham, K. & Herz, R. S. The hippocampal complex of food-storing birds. *Brain Behav. Evol.* **34**, 308–317 (1989).
- Roth, T. C. & Pravosudov, V. V. Hippocampal volumes and neuron numbers increase along a gradient of environmental harshness: A large-scale comparison. *Proc. Biol. Sci.* **276**, 401–405 (2009).
- Barnea, A. & Nottebohm, F. Seasonal recruitment of hippocampal neurons in adult free-ranging black-capped chickadees. *Proc. Natl. Acad. Sci. USA* **91**, 11217–11221 (1994).
- Sherry, D. F., Grella, S. L., Guigueno, M. F., White, D. J. & Marrone, D. F. Are there place cells in the Avian Hippocampus?. *BBE* **90**, 73–80 (2017).
- Payne, H. L., Lynch, G. F. & Aronov, D. Precise spatial representations in the hippocampus of a food-caching bird. *bioRxiv* (2020).
- Clayton, N. S. & Dickinson, A. Episodic-like memory during cache recovery by scrub jays. *Nature* **395**, 272–274 (1998).
- Paar, C. & Pelzl, J. *Understanding Cryptography: A Textbook for Students and Practitioners*. (Springer Science & Business Media, 2009).
- Barnea, A. & Nottebohm, F. Patterns of food storing by black-capped chickadees suggest a mnemonic hypothesis. *Animal Behav.* **49**, 1161–1176 (1995).
- Weinberger, K., Dasgupta, A., Langford, J., Smola, A. & Attenberg, J. Feature hashing for large scale multitask learning. in *Proceedings of the 26th Annual International Conference on Machine Learning* 1113–1120 (ACM, Montreal Quebec Canada, 2009). <https://doi.org/10.1145/1553374.1553516>.
- Cristianini, N. & Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. (Cambridge university press, 2000).
- Duda, R. O., Hart, P. E. & Stork, D. G. *Pattern Classification* 2nd edn. (Wiley-Interscience, 2000).
- Omer, D. B., Maimon, S. R., Las, L. & Ulanovsky, N. Social place-cells in the bat hippocampus. *Science* **359**, 218–224 (2018).
- Ganguli, S. & Sompolinsky, H. Compressed sensing, sparsity, and dimensionality in neuronal information processing and data analysis. *Annu. Rev. Neurosci.* **35**, 485–508 (2012).
- Petrantonakis, P. C. & Poirazi, P. A compressed sensing perspective of hippocampal function. *Front. Syst. Neurosci.* **8**, (2014).
- Maoz, O., Tkačik, G., Esteki, M. S., Kiani, R. & Schneidman, E. Learning probabilistic neural representations with randomly connected circuits. *Proc. Natl. Acad. Sci.* **117**, 25066–25073 (2020).
- Ganguli, S. & Sompolinsky, H. Short-term memory in neuronal networks through dynamical compressed sensing. *Adv. Neural Inform. Process. Syst.* **23**, (2010).
- Wolfson, H. J. & Rigoutsos, I. Geometric hashing: An overview. *IEEE Comput. Sci. Eng.* **4**, 10–21 (1997).

Acknowledgements

This work was supported by the Israeli Science Foundation (ISF; Grant 2505/20).

Author contributions

OF developed the theory and simulations. SM provided the biological background and rationale as well as contributed to the simulations and analysis.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to O.F.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-

NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024