Flexible representation and memory of higher-dimensional cognitive variables with grid cells

Mirko Klukas [*1], Marcus Lewis[1], and Ila Fiete[2]

[1]Numenta, Inc., Redwood City, CA 94063
[2]Department of Brain and Cognitive Sciences, MIT, Cambridge, MA 02139

Abstract

Grid cell representations are simultaneously flexible and powerful yet rigid and constrained: On one hand, they can encode spatial or a variety of non-spatial cognitive variables (Constantinescu et al., 2016; Killian et al., 2012), with remarkable capacity, integration, and error correction properties (Fiete et al., 2008; Sreenivasan and Fiete, 2011; Mathis et al., 2012). On the other, states within each grid module are confined to a fixed two-dimensional (2D) set across time, environment, encoded variable (Yoon et al., 2013, 2016), behavioral states including sleep (Gardner et al., 2017; Trettel et al., 2017), with the inherent low-dimensionality etched directly into the physical topography of the circuit (Heys et al., 2014; Gu et al., 2018). The restriction to 2D states seemingly imposes a severe limit on the representation of general cognitive variables of dimension greater than two by grid cells. We show here that a set of grid cell modules, each with only 2D responses, can generate unambiguous and high-capacity representations of variables in much higher-dimensional spaces. Specifically, M grid modules can represent variables of *arbitrary* dimension up to $2M$, with a capacity exponential in M. The idea generalizes our understanding of the 2D grid code as capable of flexible reconfiguration to generate unique high-capacity metric codes and memory states for representation and algebra in higher-dimensional vector spaces, without costly higher-dimensional grid-like responses in individual cells.

1 Introduction

Entorhinal grid cells in mammals are believed to play a central role in the representation of spatial information. However, unlike a hippocampal place cell, which fires at a specific location in an environment, a grid cell's spatial response is periodic and thus highly ambiguous. The representational power of grid cells emerges on the population level, or more specifically, at the level of a population of populations: Grid cells are clustered into *modules*, or networks of cells with a common spatial period. Each module specifies one's location in two spatial dimensions as a periodic two-dimensional (2D) phase, which is still highly ambiguous. But the combination of different modules breaks the degeneracy and permits the unique representation of exponentially large amounts of space (or equivalently, of a large number of distinct spaces (Fiete et al., 2008)) at fixed resolution as a function of the number of modules and thus neuron number (Fig. 1a), even if the different spatial periods are not too different in size (as in the data, where the range of periods is $\approx 0.3 - 1.2$ m), an extraordinary feature of the grid cell code relative to classical population codes (Fiete et al., 2008; Sreenivasan and Fiete, 2011).

Grid cells seem to exhibit a great deal of flexibility in their ability to also represent cognitive variables other than one's own location in 2D physical space. They respond to the location of visual gaze on a 2D image (Killian et al., 2012; Nau et al., 2018; Julian et al., 2018), the locus of covert attention in 2D space (Wilming et al., 2018), or the values of two parametrically varied features of cartoon bird images (Constantinescu et al., 2016). In all these cases, the recorded cells exhibit a response structure that matches that of grid cells during 2D movement, with single unit recordings in rodents indicating that the same grid cells are reused across variable types (Kraus et al., 2015; Aronov et al., 2017), suggesting that all of these variable types are represented by a single population of grid cells (Fig. 1c).

---

*mirko.klukas@gmail.com
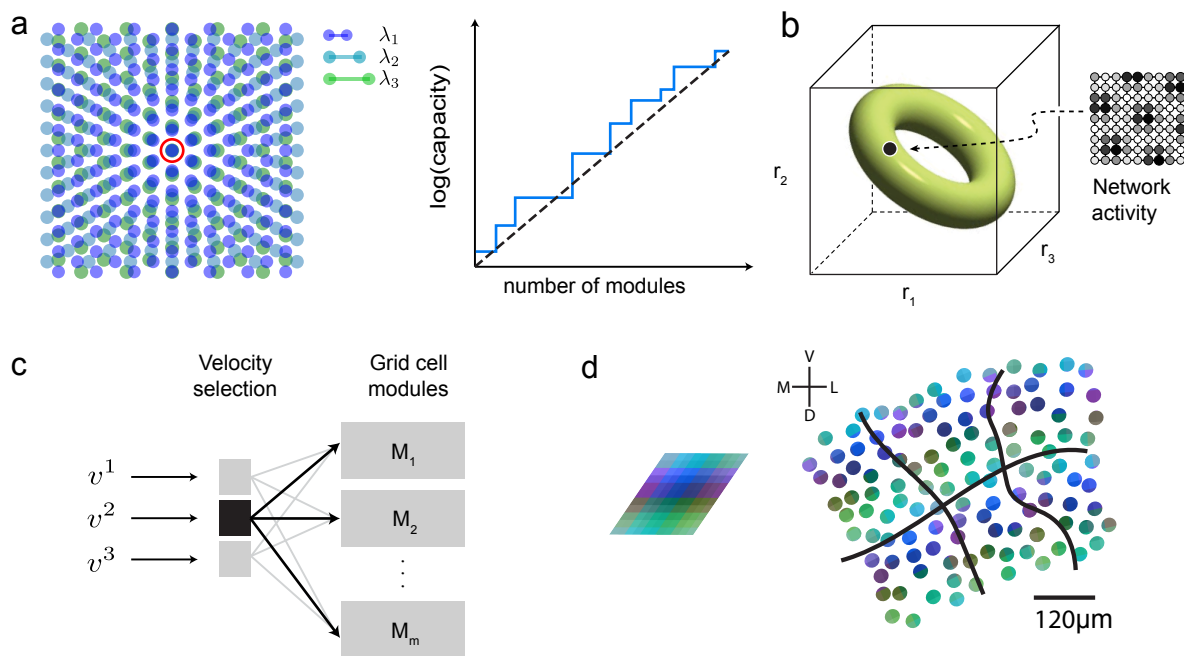
Figure 1: **Unique properties of coding with grid cells.** **(a)** Left: The combined activity of multiple modules (here, three) resolves the highly ambiguous periodic responses of a single module, over ranges that exceed the individual periods (Fiete et al., 2008). Right: The unique coding range or capacity of the multi-module code grows exponentially with the number of modules (schematic) (Fiete et al., 2008; Sreenivasan and Fiete, 2011). **b)** The activity of the thousands of grid cells within a module is hypothesized by dynamical models to be confined to the surface of a 2D torus in the high-dimensional state space (activity space) (Burak and Fiete, 2009); each point on the torus is one phase of the population-level activity pattern in the module (inset); different points are rigid 2D shifts or translations of this periodic pattern. **(c)** A velocity-selection mechanism to control which of multiple different velocity signals (here, $v^1, \ldots, v^3$) the grid cell circuit receives and processes (the dark box represents the selected signal which is allowed to pass through), can enable flexible representation of different cognitive variables. **(d)** The physical layout of grid cells in the cortical sheet (right; axes are dorsal-ventral (D,V) and medial-lateral (M,L), cells color-coded by their spatial firing phase in 2D environments, according to inset on left) matches the 2D spatial tuning phases of cells relative to each other in 2D environments.

At the same time, grid cell responses are highly rigid: Across a range of novel, familiar, and distorted spatial environments (Yoon et al., 2013), during navigation on sloped terrains (Hayman et al., 2011, 2015) or reduced-dimensional 1D tracks (Yoon et al., 2016), and most strikingly, across sleep states when the animal receives no external spatial inputs and is rather driven by presumably high-dimensional internal spontaneous activity (Gardner et al., 2017; Trettel et al., 2017), the states of grid cells are confined to a fixed set of states with preserved pairwise cell-cell correlations that match those measured during awake exploration in familiar 2D spatial environments. The fact that across coding states including sleep, grid cells conserve the pairwise firing relationships they exhibited in their 2D spatial responses directly show that the dynamics of a grid module and thus its representational power is confined to an inherently determined 2D set of states that is invariant across time, task, and behavioral state.

Even the physical layout of grid cells in the brain mirrors, and likely drives, the rigid 2D nature of the functional response of grid modules. Grid cells are spatially organized in a 2D grid-like topographical pattern based on their spatial tuning in 2D environments (Heys et al., 2014; Gu et al., 2018). Thus, the intrinsically 2D nature of the grid cell representations across diverse tasks and variables is not only functional, but physical.

These findings are puzzling together: The flexibility of mammalian grid cells in representing cognitive variables apart from one's spatial location in 2D space (Killian et al., 2012; Constantinescu et al., 2016), together with the special theoretical properties of the grid code for efficient, robust, and unique representation of a very large number of states (Fiete et al., 2008; Sreenivasan and Fiete, 2011), suggest that it might underlie very general types of cognitive representation. On the other hand, cognitive variables aren't limited to 2D (Fig. 2),

2

so the structural and dynamical constraints on grid cell activity seem severely limiting. How generally useful can the grid code be, if the autonomous states of each grid module are inherently two-dimensional? We consider what kinds and dimensions of variables, theoretically, it is possible for grid cells to represent.

The research field has reached for solutions to this problem by searching for higher-dimensional grid patterns on higher-dimensional tasks. In some 3D navigation experiments the recorded grid cells fire along the third dimension without a change relative to their 2D responses, and seemingly convey no information about the third dimension (Hayman et al., 2011). In other experiments, grid cell responses might show some modulation in the third dimension (Jeffery and Grieves, 2018; Ginosar et al., 2018) however, the nature and structure of the responses and response dimension are not fully elucidated. Moreover, given the strong evidence described above about the inherent drive towards 2D dynamics in the grid cell circuit of at least rats even during sleep, it is unclear to what extent the responses (could) resemble a higher-dimensional grid. Finally, as considered in the Discussion, the construction of high-dimensional grids is extremely costly in principle, requiring exponentially more neurons as a function of dimension, at least in attractor network models.

In the present work, we propose a different solution to the encoding of high-dimensional information by grid cells, even as they retain a rigid 2D structure in each module. We show that the power of the multi-module grid cell representation extends far beyond variables of two dimensions, allowing grid cells to generate unique and unambiguous codes for variables of dimension up to twice the number of modules. Moreover, we show that the coding range at fixed resolution along each dimension remains exponential as in 2D, even when representing high-dimensional variables.
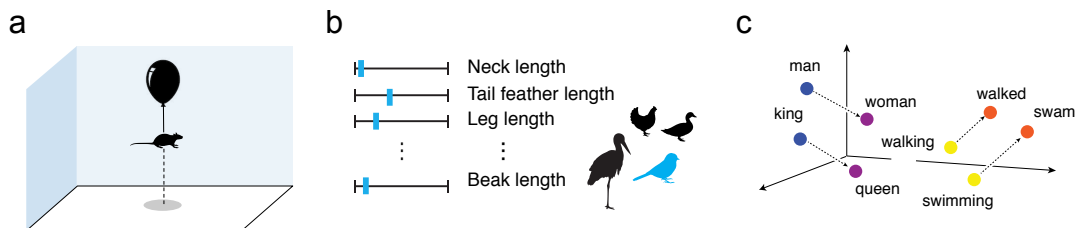

Figure 2: **The need for higher-dimensional cognitive representations.** **(a)** An animal's location in 3D space (especially relevant for flies, birds, fish, bats). **(b)** A multi-dimensional cognitive space consisting of cartoon birds with multiple characteristics that are each are parametrically varied, extending the 2D "stretchy birds" map of Constantinescu et al. (2016). **(c)** Semantic embeddings of words into a Euclidean space (right) that respects semantic similarity and encodes grammatical and semantic relationships (Mikolov et al., 2013).

2 Results

2.1 The high-capacity grid code for self-location in 2D

A mammalian grid cell is defined by the periodic arrangement of its firing fields on the vertices of an equilateral triangular lattice that tiles 2D environments explored by the animal. Grid *modules* are discrete sub-networks of grid cells with a common lattice period and orientation but uniformly distributed 2D translational offsets.

As defined in (Fiete et al., 2008), a grid module represents the animal's time-varying location $x(t)$ by *a macroscopic periodic variable*, a 2D phase $\phi(x(t))$ with respect to that module's periodic response and relative to a reference phase: if the animal moves by integer multiples of the lattice period along either or both of the two primary lattice axes, the module response repeats and is thus is unchanged.

A grid module, because of its periodic response, provides a non-unique representation of 2D locations. However, as shown previously (Fiete et al., 2008; Sreenivasan and Fiete, 2011) and summarized in Figure 1a, the existence of multiple modules with distinct (though similarly sized) periods means that the representational capacity of grid cells – that is, their ability to generate unique and well-separated coding states for different locations – grows exponentially in the number of modules (and thus number of total grid cells), while maintaining a fixed resolution.

2.2 Velocity integration for flexible and instantaneous representation of arbitrary 2D variables

Central to key (attractor) models of grid cell dynamics (Burak and Fiete, 2009), grid cell states are updated based on estimated *changes* in animal position. Specifically, the module phase is changed in proportion to the animal's estimated instantaneous velocity ($\hat{v}$):

$$\frac{d\phi}{dt} = A \cdot \hat{v}. \tag{1}$$

Here $A \in \mathbb{R}^{2 \times 2}$ is a linear velocity projection operator that governs how the animal's movements in the 2D world map to changes in the internal module phase.

Once the module is anchored, by assignment of a particular phase φ_0 to a specific position x_0 in the environment (e.g. through place cells or other spatially-specific cells (Welinder et al., 2008; Burgess, 2008)), this velocity-based updating will automatically generate a grid code (phase) for any other location x in the environment reached via the path $\gamma(x_0, x)$:

$$\phi(x) = \varphi_0 + \int A \cdot \frac{d\gamma}{dt} \, dt \pmod 1 \tag{2}$$

If the locations x lie in Euclidean space, the assigned phase is guaranteed to be independent of the particular path $\gamma(x_0, x)$ taken between x_0 and x, thus ensuring a well-defined grid code for x regardless of trajectory:

$$\phi(x) = \varphi_0 + A(x - x_0) \pmod 1, \tag{3}$$

Under the attractor models, the different periods across modules could be generated by simple rescalings of the velocity projection matrix A. In this view, the projection matrix A_α for module α is given by

$$A_\alpha = g_\alpha A, \tag{4}$$

where g_α is a simple scalar gain factor. When a given module rescales its response due to environment effects (Barry et al., 2007), the attractor model predicts that the rescaling *must* be generated by a gain change in the velocity projection (Burak and Fiete, 2009).

We highlight two remarkable theoretical attributes of this velocity-based construction of a map between the external variable x and the grid code $\phi(x)$: 1) The grid modules will, without reconfiguration, integrate *any* (Euclidean) 2D velocity input to produce a unique and well-defined grid code; thus, if a different velocity signal, related to changes in any other variable, is fed into the circuit, the modules will now generate a grid code for that variable. 2) Because the mapping between values of the external variable to grid phases is automatically and instantaneously defined starting from a single anchor point, through the velocity mapping mechanism, it does not require the slow and painstaking construction of a point-by-point external state-to-internal state feedforward correspondence or map for each value that the external variable can take.

Property 1) above, with a velocity signal related to the rate of change of gaze direction, auditory pitch, locus of attention, or navigation through a parametric space of images, supplied as input to the grid circuit, would produce a 2D grid as a function of the integral of that 2D velocity. This is compatible with, and explains, recent empirical studies in which grid cells appear to readily display periodic coding during navigation through various continuous one- and two-dimensional cognitive spaces (Killian et al., 2012; Nau et al., 2018; Julian et al., 2018; Constantinescu et al., 2016; Wilming et al., 2018). The key additional piece of machinery required for flexible and generalizable grid representation of 2D cognitive variables is a switching mechanism within the circuit that supplies velocity inputs to the grid cells. The role of the switch is to gate which of many potential velocity signals are allowed to drive the grid modules, Figure 1b.

2.3 Ambiguous representation of N-D variables by 2D grid cells

If grid cells cannot generate higher-dimensional grid responses, one way for them to generate a response while navigating through higher-dimensional cognitive spaces is to simply project the N-dimensional velocity to two dimensions, accomplished by a $2 \times N$ velocity projection operator A (in place of the previous 2×2 projection).

As a result, grid cells would faithfully integrate motion along the two projected dimensions, generating 2D grids, but ignore motion along the remaining dimensions, generating elongated and undifferentiated responses along those dimensions. This scheme is consistent with the empirical findings in Hayman et al. (2011).

However, the projection or compression of displacement information from N to 2 dimensions means that $N-2$ of the components of the N-dimensional velocity project to zero under A, resulting in huge information loss. The resulting 2D grid representation is ambiguous: it is not one-to-one decodeable over any range of x, including within the smallest module period, Figure 3. The larger the dimension N, the greater the ambiguity and information loss.
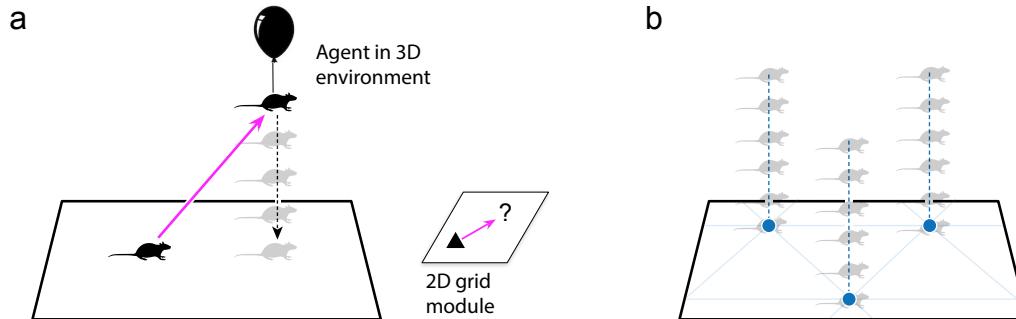

Figure 3: **Double ambiguity from projection of aperiodic higher-dimensional input onto 2D periodic responses. (a)** If the encoded variable is 3D (here, the animal leaves the 2D plane), simple projection down to the 2D phase is ambiguous and consistent with multiple locations in the z-direction, so it is unclear how to update the 2D phase in the module. **(b)** The periodicity of the grid code in a module produces a different ambiguity, this one within in the x-y plane.

2.4 Unique representation of N-D variables by multi-modular 2D grid cells

The central observation of this paper is that just as the existence of multiple modules operating independently to integrate velocity solves the problem of the ambiguity of representation by periodic responses, it can also *simultaneously* solve the ambiguity that results from the compression of higher-dimensional inputs to two-dimensional responses.

We propose that the grid cell system might uniquely encode higher-dimensional variables by constructing M distinct $(2 \times N)$ operators A_α that project velocities in N-dimensional spaces into 2D velocity signals for each of the M grid modules. These signals are not related by simply a scalar gain, but have to differ more fundamentally in their responses to each input dimension (Fig. 4). In consequence they compress different portions of the input space and can mutually resolve their ambiguities (Fig. 4).

A set of M independent projections can be viewed as a single matrix of size $2M \times N$, and this matrix is full-rank if $M \geq N/2$ (or equivalently, $N \leq 2M$) and its columns are linearly independent. In other words, the intersection of the kernels of all projections must be trivial, consisting solely of the zero vector. The combined projection is in fact of full rank with high probability if the projections are chosen randomly, when $N \leq 2M$ (Appendix). Ignoring for the moment the periodic nature of the grid cell code, or equivalently considering a range that is smaller than the smallest grid period (per dimension), a set of M random projections from an N-dimensional input to a set of 2D inputs thus preserves information when $2M$ is greater than or equal to N.

Next, this set of randomly-projected 2D inputs, each driving a single grid module with periodic output, forms an invertible mapping over a range of the inputs that can exceed the individual grid periods if $2M > N$. This is true for the same reason that combining periodic grid responses at multiple spatial scales gives rise to an invertible mapping from non-periodic 2D variables to a set of 2D phases (Fiete et al., 2008), Figure 1a. In sum, the mapping $\phi(x)\colon \mathbb{R}^N \to T^{2M}$, where T^{2M} a 2M-dimensional torus defined by the 2D phases of M grid modules, constitutes an invertible and unique code of the N-dimensional variable x over ranges bigger than the periods of any individual modules.

With realistic estimates for the number of grid modules in an individual animal ($M = 4, \ldots, 8$), the code could represent variables that lie in quite high-dimensional spaces ($N = 8, \ldots, 16$).

In sum, supplied with a flexible velocity input switching mechanism, grid cells provide a relatively high-dimensional neural affine vector space[1] for memory, integration, and representation of N-dimensional cognitive

---

[1]The term *affine* makes explicit the lack of a preferred "zero" element. Each point in the space admits a neighbourhood that

variables when $N \leq 2M$, and over ranges larger than the scales of the individual grid periods. It is a memory because when velocity inputs are zero, the state persists. In this vector-like space, the mathematical operations of vector addition or vector displacements are easily performed.
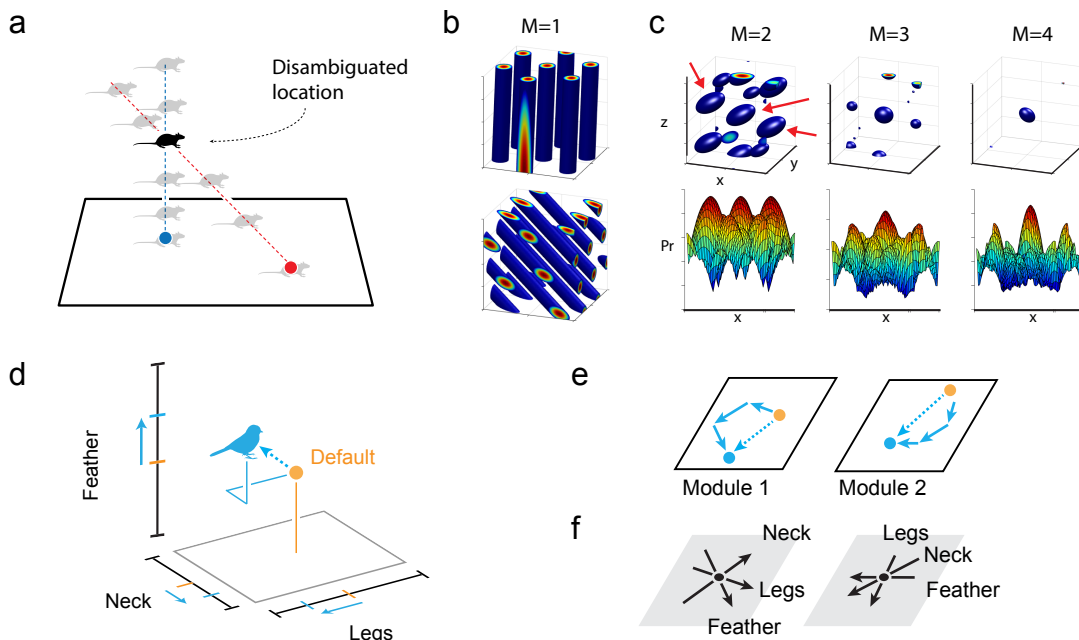
Figure 4: **Solution to the double ambiguity of grid cell representation: a three-dimensional example. (a)** Two different 2D phases for two modules are set by two distinct projections (red and blue) of the 3D value onto a plane. **(b)** The 3D tuning curves of two cells from different modules, based on the 2D phases in (a). Note that the responses are doubly ambiguous, with ambiguity from the periodic nature of the response in two dimensions the extended nature of the response in the remaining $N-2$ (here, 1) dimension. **(c)** Top row: Estimates of the value of the encoded 3D variable obtained by combining the highly ambiguous estimates of 2, 3, and 4 cells as in (b). Given the cell responses we compute a probability estimate and show only areas that exceed a fixed threshold (blue blobs). Bottom row: We show the probability estimate for the x-coordinate only. For $M = 2$ the probability distribution has three maximal isolated peaks (corresponding blobs marked with red arrow), meaning that the combined code has resolved *all* ambiguity in 3D between the maximal peaks. The spacing between the maximal peaks is the capacity, or range, over which the code is unique. With more modules, as shown in the remaining columns $M = 3, 4$, the range quickly grows larger than the individual periods. **(d)** A conceptual view of bird space from Figure 2 showing three dimensions only. A specific instance of a bird (blue) can be viewed in comparison to a default bird (orange). It can be described as a list of "offsets" (blue vectors) from the default value (orange marker) in each dimension. **(e,f)** The offsets are translated into a distinct 2-dimensional "phase shift" (blue vectors) on each module (black rhombi) that can be applied to a default phase (orange dot). The resulting phases (blue dots) yield a unique grid code for the blue bird in (e). The projections are determined by the images of the axes in bird space onto phase shifts which are shown in (f).

2.5   The capacity grows exponentially with module number

The capacity of the 2D grid code for 2D variables is extremely large, growing exponentially with the number of modules (Fiete et al., 2008; Sreenivasan and Fiete, 2011; Mathis et al., 2013), which serves the essential role of generating a large numbers of unique and well-separated neural representations. By devoting multi-module coding to the problem of generating unique codes for high-dimensional variables, is the high capacity lost?

---

naturally carries the structure of a vector space with the point at its origin. Note that the space is a quotient of Euclidean space with a lattice.
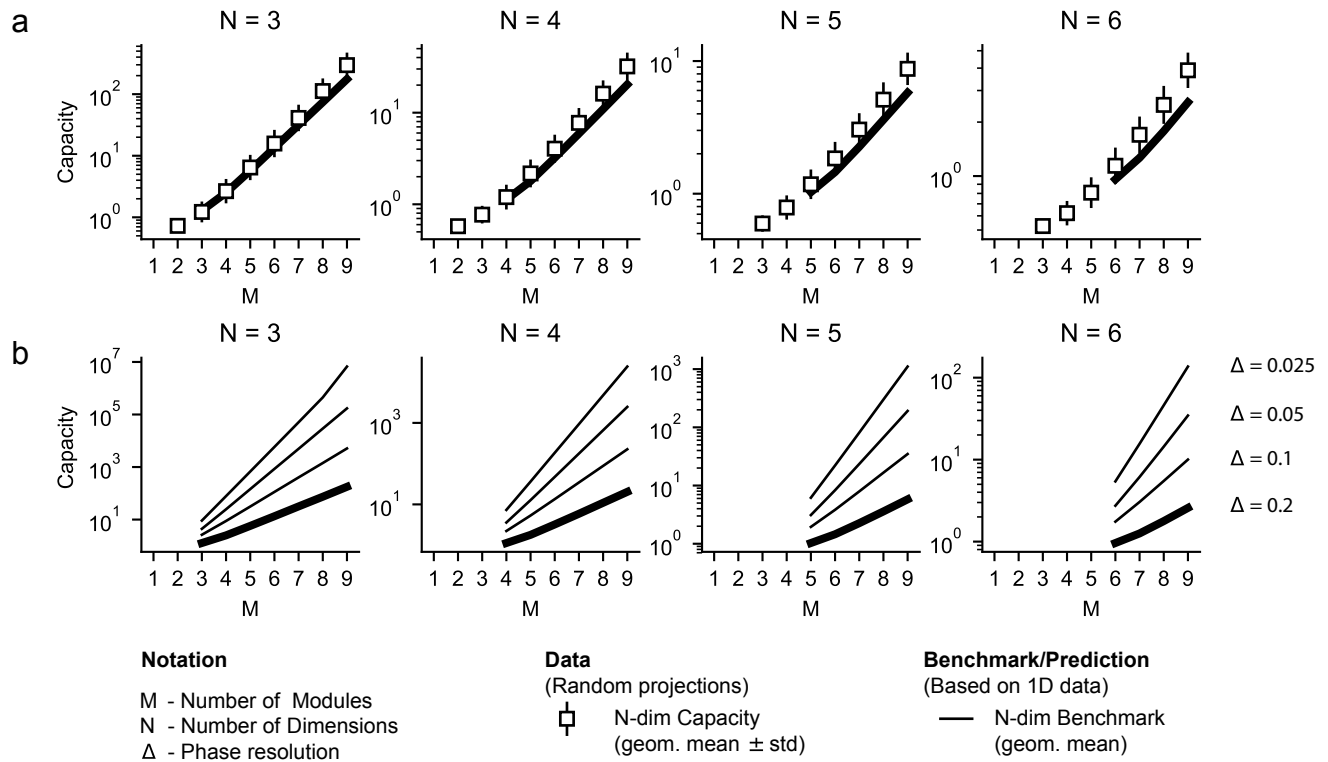
Figure 5: **Capacity grows exponentially with module number.** Capacity is given in multiples of the mean grid period: for an average period of 0.5m, the capacity is in units of a half-meter. **(a)** Exact coding range of the grid code for variables of dimension 3 to 6, assuming an overly conservative phase resolution of $\Delta = 0.2$ to reduce time of computation. We show the geometric mean and standard deviation over 1000 different draws of the projection matrices A for each pair M, N. The module periods were normally distributed around 1.0 with standard deviation 0.2, and each module's projection was chosen semi-randomly (see Methods). To compute the expected value $\mathbb{E}(W)$ of the benchmark in Equation 5, we also run this simulation with $N = 1$ (Appendix, Fig. 7), solid line. The capacity grows exponentially with the number of modules; the benchmark provides an estimate of the expected capacity. **(b)** We use the benchmark to show the coding range for more realistic values of phase resolution ($\Delta = 0.2, \ldots, 0.025$). We chose the benchmark rather than measuring the exact range for practical reasons (the run-time scales with the *volume* of the coding range not its side-length). Results shown for $M/N \geq 1$.

To address this question, we characterize the range over which the higher-dimensional grid cell code is unique and invertible – its *capacity*, as in (Fiete et al., 2008). We define capacity to be the maximal side-length of a hypercube of dimension N over whose volume no two points are assigned the same grid code.

To set a rough benchmark for the capacity of the coding scheme, we define a conceptually simpler yet effective coding scheme for a variable x of dimension N (with $2 < N \leq 2M$): For simplicity assume that M is a multiple of N. Then we can divide the collection of modules into N distinct groups, each consisting of M/N modules. Let each group of M/N modules encode one of the N coordinates of x. Thus, the problem of encoding a variable of dimension N has decomposed into the task of encoding N variables of dimension 1 each. The capacity W will be determined by be the minimum capacity of one of the groups, that is

$$W = \min\{W_1, \ldots, W_N\}, \tag{5}$$

where W_i ($i = 1, \ldots, N$) denotes the $1D$-capacity of the ith group. Keep in mind that we are choosing the projection matrices in a random fashion and can view the W_i, and in turn W, as random variables. We know from previous theoretical work (Fiete et al., 2008) that the capacity of encoding a 1D phase with a number of 2D modules grows exponentially in the number of modules. Thus, the capacity of each group increases with the number of modules per group, or as M/N. More precisely, as in (Fiete et al., 2008), for a phase resolution

(phase bin size) Δ, the expected capacity $\mathbb{E}(W_i)$ of each group will scale as the number of distinct phases times the bin size (cf. Appendix, Figure 7b), i.e.

$$\mathbb{E}(W_i) \propto \Delta \Big(\frac{1}{\Delta^2}\Big)^{M/N}. \tag{6}$$

If M is a multiple of N we can actually collect data and compute the empirical distribution for W_i (Appendix, Figure 7a) and in turn can compute an expected value for W (Fig. 5, solid lines). For non-integer values $M/N \geq 1$ we can either interpolate between existing data points, or interpolate between the distributions of W_i. This benchmark sets our expectations for how capacity could scale, and suggests that it should, as in the 2D case, increase exponentially with the number of modules even when encoding higher-dimensional variables.

Next, we numerically assess the performance of our proposed random projection-based grid cell coding scheme for higher-dimensional variables. For input variables of various dimensions ($N = 3, \ldots, 6$), we generate random projection operators and numerically compute the capacity as a function of the number of encoding grid modules ($M = 1, \ldots, 9$); white squares in Figure 5a show the mean value. The capacity of the randomized scheme clearly grows exponentially, and further, the rate of exponential growth closely matches the benchmark. In sum, the 2D grid code with multiple modules is capable not only of unambiguous representation of arbitrary 1D and 2D variables with exponentially large capacity as a function of number of modules, but it can do the same for much higher-dimensional variables as well.

We also consider and show how capacity changes with the phase resolution within each module ($\Delta = 0.4, \ldots, 0.025$) (Appendix, Figure 8). Consistent with (Fiete et al., 2008), capacity grows as a power of phase resolution, regardless of the dimensionality of the encoded variable.

2.6 Predicted tuning curves for N-dimensional representations
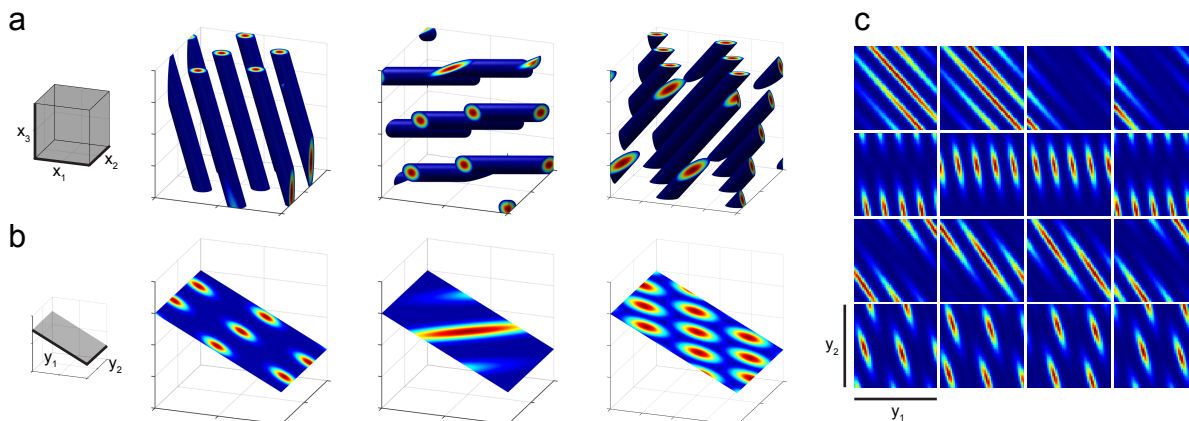
Figure 6: **Predictions about grid cell firing and cell-cell relationships in higher dimensions.** For ease of illustration, we consider here the encoding of a variable in three dimensions. **(a)** Left: The 3D space of the encoded variable. Remaining panels: 3D tuning curves of three cells from different modules. **(b)** Left: the gray plane is an arbitrarily chosen 2D subspace of the 3D variable; we will consider how neural responses vary as the input varies along this 2D subspace. Remaining panels: Cross-sections of the tuning curves of the same cells shown in (a), along the selected 2D subspace of the input variable. **(c)** Top row: 2D input subspace responses (as in (b)) of four cells from a common module are closely related to, but not simple translations of, each other. Remaining rows: sets of 4 co-modular cells, different rows are different modules.

We have shown theoretically that it is possible for grid cells to quite simply, and without reconfiguration, represent higher-dimensional variables. Will it be possible to identify whether the brain could be exploiting this functionality?

A key signature of non-degenerate higher-dimensional encoding through our proposed scheme of independent low-dimensional grid cell projections lies in the differences in predicted tuning curves across modules, described next.

8

When an N-dimensional input variable is projected to 2D to drive a single module, then changes of the variable along the $(N-2)$-dimensional null space of the projection result in no change to the module state. Thus, the N-dimensional tuning curves of the module will always look like "lifts" (unchanging responses) along these $N-2$ dimensions, of a response that is grid-like in some 2D planar slice. If the input variable is 3-dimensional, these lifts simply look like elongated bands or pillars (see Figure 6b, left panel)[2]. These $N-$dimensional responses are predicted to differ across modules because of the different velocity projections in our theory (different examples in Figure 6a).

While it is straightforward to state the predicted properties of neural responses to N-dimensional variables, it is difficult to record, visualize, and characterize N-dimensional tuning curves in both principle and practice. Fortunately, doing so is not necessary to test the above prediction: It is actually sufficient to simply plot and characterize the responses of cells in different modules as a function of variations along any 2D subspace of the explored N-dimensional input space.

A 2D subspace of the N-dimensional input space will intersect with the different N-dimensional tunings of the modules to produce different 2D responses across modules. There are two cases to consider: In the first and atypical case, the selected 2D subspace happens to exactly align with one of the null or lift directions of a module. As the input is varied along both dimensions of the 2D subspace, this module's response only changes along one of them, and thus grid cells in the module will have periodically arranged stripe responses (Figure 6b, middle panel).

In the second and typical case, the 2D subspace is not exactly aligned to one of the null directions of the module, so that variations in the input space along the 2D subspace have some non-zero projection onto each of the two non-null grid-like response dimensions of the N-dimensional tuning curves. The responses of grid cells along the 2D subspace will then resemble distorted (not necessarily equilateral triangular) grids (Figure 6b, left and right panels). In fact, the responses of modules in this case can range from perfectly equilateral triangular grids (Figure 6b, left panel) to non-grid-like and relatively complex, resembling bands of bands (Figure 6c). For a broader sampling of possible grid cell response geometries, see Appendix Figure C.

Within a module, the responses of different cells are generated from translations of the N-dimensional tuning of the module, and are predicted to exhibit systematic relationships (each row of Figure 6 shows co-modular responses). When plotted over a large enough area, these 2D responses are translations of each other; but when plotted over smaller areas, they may not appear as simple as shifts of a canonical 2D response pattern (e.g. Figure 6c, top row), similar to the relationships seen in co-modular cells in 1D environments which are generated by cutting a lower-dimensional slice through translations of a higher-dimensional (2D) lattice (Yoon et al., 2016).

The structured response relationships of co-modular cells in our model are a simpler prediction to test than across-module diversity in tuning, because of the relative ease of simultaneously recording multiple co-modular cells.

In sum, a central prediction of the hypothesis that the grid cell system could collectively use its multiple modules to encode variables of higher dimension than two is that the projections to different modules should be different, and therefore that in such situations, the responses of grid cells in different modules will differ in the geometry of their tuning curves.

3 Discussion

In sum, supplied with a flexible velocity input switching mechanism, grid cells provide a neural vector-like space for integration, memory, and representation of 2D or higher-dimensional variables. It is a memory because when velocity inputs are zero, the previous state persists. In this vector-like space, the mathematical operations of vector addition or vector displacements are easily performed.

3.1 Summary of key findings

We have presented a conceptually new possibility for the grid code: that a small collection of (M) grid cell modules can collectively provide unique representations for higher-dimensional variables (of dimension $N \leq 2M$-

---

[2]Interestingly, there is always some planar slice of the N-dimensional tuning curve along which the response will be a 2D equilateral triangular lattice regardless of anisotropies in the 2D projection of the velocity projection operator A (Appendix; this is true whenever A is of rank 2, which is always the case for random choices except for special cases that makeup a set of zero measure (probability tending to 0)).

D) even when the intrinsic response in individual modules is rigidly low-dimensional (2D). Moreover, the higher-dimensional representations that we show are possible have an exponentially large capacity: the range of the code increases exponentially with the number of modules (scaling with $2M/N$ in the exponent) at a fixed local resolution, extending matching theoretical results about grid cells in their ability to represent 1D and 2D variables.

3.2   Implications for computation

The multi-module or population-of-population representation of grid cells provides a pre-fabricated, general higher-dimensional neural vector space that can be used for both representation and memory of arbitrary vectors (of dimension $\leq 2M$), and more specifically, for integration of vector inputs. The update mechanism of grid cells permits vector-algebraic operations between the stored vectors, required for vector integration in higher-dimension abstract spaces. So long as displacements in the abstract spaces are provided as inputs to the network, the network can thus represent, hold in memory, and perform algebraic sum operations on general, abstract vectors without any reconfiguration of the recurrent grid cell network.

We believe these results and implications fulfill, at least in theory, intuitive expectations that the very peculiar grid code might be extraordinary in the computations it enables.

3.3   Predictions

The key prediction of whether the brain utilizes these potential capabilities is that the projections of high-dimensional variables into the different modules are different, even as the states within each module remain low-dimensional.

This means that the tuning curves (over a 2D slice of a potentially high-dimensional input space) of cells within a module should still look like displaced versions , or different portions, of a common 2D pattern, that differs across modules. Thus tuning curves of cells in different modules are not merely scaled copies or displaced responses in other modules (Figure 6b, different rows).

A prerequisite for this possibility is that different grid modules must be capable of changing their responses independently of each other (through the action of separate velocity projection operators); tantalizing hints that this is possible appear in (Stensola et al., 2012), where different grid modules appear to rescale by different amounts in response to an environmental deformation.

3.4   Relationship with band cells

Cells whose spatial responses resemble evenly spaced parallel bands across a 2-dimensional environment are called *band cells*. They were postulated as inputs for computational models of grid cells (Burgess et al., 2007), and later observed experimentally in recordings in the parasubiculum and entorhinal cortex of freely moving rats (Krupic et al., 2012). In our model, tuning curves can be variously grid-like, distorted or stretched grid-like, band-like, amplitude-modulated band-like, or like bands of bands. The band-like responses in our model – obtained from 2D slices of N-D projected grids at different angles – are a generalization of simple band cells, and suggest that previously observed band-like responses might be a signature of a projection operator that projects a 2D or higher-dimensional input onto one of the null directions of the grid module.

3.5   Why is the per-module grid structure two-dimensional?

If grid cells can represent abstract cognitive variables of arbitrary dimension up to about 10, why are single modules set up to be specifically and rigidly 2D? The evolutionary origins of the grid cell system might lie in spatial representation in animals largely confined to 2D spatial behaviors, thus 2D might be primal. Even so, starting from an evolutionary ancestor 2D grid module representations, there are two distinct options for branching out to generate representations of variables of dimension $N(>2)$: One is to maintain a 2D response per module but evolve the N-dimensional representation across modules, as we have proposed. The alternative is for each module to evolve an N-dimensional rather than 2D grid through its dynamics.

This latter possible strategy is the current conceptual focus of the field in studying whether grid cells could represent higher-dimensional spaces. However, it is far more costly: the number of neurons needed to build an N-dimensional grid with the same resolution as in 2D is very high: it increases exponentially with dimension, as $K^{N/2}$, where K is the number of cells used in the 2D module. By contrast, in our proposal there is no added

cost in neurons for representing higher-dimensional variables, and the cost of a shrunken representational range simply reduces the still-exponential rate at which the capacity grows with number of modules.

3.6   Observed 3D responses in grid cells

In some studies of animals exploring higher-dimensional spaces, specifically 3D spatial environments, the response of grid cells is elongated and undifferentiated along one dimension, while remaining grid-like in the other two (Hayman et al., 2011). This kind of tuning is consistent with our prediction, and we have shown it allows for unique coding along the third dimension if the projections (and thus the undifferentiated direction) are not aligned across modules.

Recently, grid cell responses have been examined in bats flying through 3D environments (Ginosar et al., 2018). Bats crawling on 2D surfaces exhibit the same 2D triangular grid cell tuning (Yartsev et al., 2011) as rats and mice. In 3D, consistent with our theory, the responses seem not to clearly exhibit regular 3D grid patterns (Yartsev and Ulanovsky, 2013). However, the fields do seem to be localized in all 3 dimensions, at least in the vicinity of a tree around where the bats forage for food (Ginosar et al., 2018). It is possible in this case that localized higher-dimensional fields are formed in the hippocampus or the lateral entorhinal cortex, based on spatial landmarks. Localized fields in 3D could also be formed by superpositions of grid cells encoding higher-dimensional spaces according to our model: if some cells in entorhinal cortex perform a readout of two or three modules, these conjunction-forming cells would exhibit localized 3D fields with some regularity in spacing (as in Fig. 4c) but not full grid-like periodicity and thus no notion of a spatial phase. A similar situation might hold for the observed localization of fields in 3D, in rats navigating 3D wire mesh cubes (Jeffery and Grieves, 2018).

However, the absence of a band-like structure in grid cells along any dimension during 3D coding is not consistent with our theory.

As noted already, local single-bump representations are too costly in the number of required neurons even in 2D spaces to have enough capacity to cover behaviorally large enough spaces with realistic numbers of neurons (Fiete et al., 2008); this problem is exponentially worse in higher dimensions, with the number of needed cells growing as the dimensionality of the space in the exponent. This simple theoretical counting argument suggests that it is simply impossible for 3D and higher-dimensional spaces of appropriate size to be tiled by a local-bump code as sometimes ascribed to place cells in the hippocampus. Instead, place cell representations will likely be found to be reserved for specific sites of especial interest in the space (locations of cues, rewards, etc.), or be found to tile spaces only when the explored space is artificially small, as is often the case in electrophysiology experiments. Thus, the only way to properly understand the true structure of neural representations for spatial and non-spatial variables of dimension greater than or equal to two is to study them when animals are allowed to explore realistically many or large expanses of these spaces, a point emphasized earlier (Fiete et al., 2008).

4   Methods

4.1   Capacity computations

In each trial we assigned each module an orthogonal projection onto a random 2D plane tiled by a randomly oriented hexagonal lattice. The spatial periods were chosen from a normal distribution around 1.0 with standard deviation 0.2, and in each trial the modules were normalized so that their mean spatial period was 1.0. In each trial if the set of projections didn't achieve a resolution of at least 0.5 units along each dimension, we selected a new set and tried again (Appendix).

Given a collection of M grid modules and their associated projection matrices, we want to determine the maximal range over which the code is unique.

We say that two points x, x' *collide* if the distance of their associated modulo phases in the grid representation is smaller than or equal to a fixed threshold $\frac{\Delta}{2}$, where Δ is the phase resolution per module. Suppose a point x is surrounded by an N-dimensional cube of side length w centered at x, and no other point in the cube collides with x. Because of the translational invariance of the grid code, any other point admits such a neighbourhood as well, and the capacity of the code is at least $\frac{w}{2}$.[3] To determine the capacity of the code it is therefore sufficient to compute the side length of a maximal collision-free cube centered at the origin.

---

[3]Note that we assumed that the cube contains no collision with its *center*. That means it could still contain another pair of colliding points. However, if it contains such a pair, these points lie at least $w/2$ units apart from each other.

In a small neighbourhood of the origin (moving along each dimension by an amount smaller than all the grid periods) the encoding map is one-to-one and continuous if the intersection of the kernels of the different projection matrices is trivial. Any point thus admits a small neighbourhood of points whose associated phases are closer than Δ; it is necessary to ignore these points while performing our search for collisions for the capacity computations.

We begin with a small N-dimensional box that encloses the ignored points, then incrementally expand the box outward. We check whether the frontier region contains any collisions by splitting it into a set of N-dimensional boxes and thoroughly checking each of these boxes for the starting point's representation following a divide-and-conquer approach. See Supplementary Material for a more detailed description of the algorithm.

4.2   Phase distance

We base our distance computations in the space of the vector of grid phases with respect to the equivalence relation defined by the lattice (each module encodes a 2D phase on a 2D torus, which we can understand as the quotient of Euclidean space with a hexagonal lattice). We now define a metric on the sets of phases by taking the maximum of their component-wise distances, i.e. for two sets of phases $\boldsymbol{\phi} = (\phi^1, \ldots, \phi^M)$ and $\boldsymbol{\psi} = (\psi^1, \ldots, \psi^M)$ we define

$$d(\boldsymbol{\phi}, \boldsymbol{\psi}) = \max\left\{ d_1(\phi^1, \psi^1), \ldots, d_M(\phi^M, \psi^M) \right\},$$

where d_i $(i = 1, \ldots, M)$ denote the metrics inherited from $\mathbb{R}^2$ with respect to each module's underlying lattices.

Acknowledgements

Competing interests

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. MK and ML were employed by Numenta Inc. Numenta has stated that use of its intellectual property, including all the ideas contained in this work, is free for non-commercial research purposes. In addition Numenta has released all pertinent source code as open source under the AGPL V3 license.

References

Aronov, D., R. Nevers, and D. W. Tank
  2017. Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit. *Nature*, 543:719.

Barry, C., R. Hayman, N. Burgess, and K. Jeffery
  2007. Experience-dependent rescaling of entorhinal grids. *Nature neuroscience*, 10:682–4.

Burak, Y. and I. R. Fiete
  2009. Accurate path integration in continuous attractor network models of grid cells. *PLoS computational biology*, 5:e1000291.

Burgess, N.
  2008. Grid cells and theta as oscillatory interference: Theory and predictions. *Hippocampus*, 18(12):1157–1174.

Burgess, N., C. Barry, and J. O'Keefe
  2007. An oscillatory interference model of grid cell firing. *Hippocampus*, 17:801–812.

Constantinescu, A. O., J. X. O'Reilly, and T. E. J. Behrens
  2016. Organizing conceptual knowledge in humans with a gridlike code. *Science*, 352:1464–1468.

Fiete, I. R., Y. Burak, and T. Brookings
    2008. What grid cells convey about rat location. *Journal of Neuroscience*, 28:6858–6871.

Gardner, R. J., L. Lu, T. Wernle, M.-B. Moser, and E. I. Moser
    2017. Correlation structure of grid cells is preserved during sleep. *bioRxiv*.

Ginosar, G., A. Finkelstein, L. Las, and N. Ulanovsky
    2018. Representation of 3D space in the entorhinal cortex of flying bats. In *47th Annual Meeting of Society for Neuroscience (SfN), San Diego*.

Gu, Y., S. Lewallen, A. A. Kinkhabwala, C. Domnisoru, K. Yoon, J. L. Gauthier, I. R. Fiete, and D. W. Tank
    2018. A map-like micro-organization of grid cells in the medial entorhinal cortex. *Cell*, 175:736–750.

Hayman, R., M. A. Verriotis, A. Jovalekic, A. A. Fenton, and K. J. Jeffery
    2011. Anisotropic encoding of three-dimensional space by place cells and grid cells. *Nature Neuroscience*, 14(9):1182–1190.

Hayman, R. M. A., G. Casali, J. J. Wilson, and K. J. Jeffery
    2015. Grid cells on steeply sloping terrain: evidence for planar rather than volumetric encoding. *Frontiers in Psychology*, 6:925.

Heys, J. G., K. V. Rangarajan, and D. A. Dombeck
    2014. The functional micro-organization of grid cells revealed by cellular-resolution imaging. *Neuron*, 84:1079–90.

Jeffery, K. and R. Grieves
    2018. Talk. Grid Cell Meeting.

Julian, J., A. Keinath, G. Frazzetta, and R. Epstein
    2018. Human entorhinal cortex represents visual space using a boundary-anchored grid. *Nature Neuroscience*, 21:191–194.

Killian, N. J., M. J. Jutras, and E. A. Buffalo
    2012. A map of visual space in the primate entorhinal cortex. *Nature*, 491:761–764.

Kraus, B., M. Brandon, R. Robinson, M. Connerney, M. Hasselmo, and H. Eichenbaum
    2015. During running in place, grid cells integrate elapsed time and distance run. *Neuron*, 88(3):578 – 589.

Krupic, J., N. Burgess, and J. O'Keefe
    2012. Neural representations of location composed of spatially periodic bands. *Science*, 337:853–857.

Mathis, A., A. Herz, and M. Stemmler
    2013. Multiscale codes in the nervous system: The problem of noise correlations and the ambiguity of periodic scales. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 88:022713.

Mathis, A., A. V. M. Herz, and M. Stemmler
    2012. Optimal population codes for space: Grid cells outperform place cells. *Neural Computation*, 24:2280–2317.

Mikolov, T., K. Chen, G. Corrado, and J. Dean
    2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Nau, M., T. Navarro Schröder, J. Bellmund, and C. F. Doeller
    2018. Hexadirectional coding of visual space in human entorhinal cortex. *Nature Neuroscience*, 21:188–190.

Sreenivasan, S. and I. Fiete
    2011. Grid cells generate an analog error-correcting code for singularly precise neural computation. *Nature Neuroscience*, 14:1330–1337.

Stensola, H., T. Stensola, T. Solstad, K. Frøland, M.-B. Moser, and E. I. Moser
    2012. The entorhinal grid map is discretized. *Nature*, 492(7427):72–78.

Trettel, S. G., J. B. Trimper, E. Hwaun, I. R. Fiete, and L. L. Colgin
2017. Grid cell co-activity patterns during sleep reflect spatial overlap of grid fields during active behaviors. *bioRxiv*.

Welinder, P. E., Y. Burak, and I. R. Fiete
2008. Grid cells: The position code, neural network models of activity, and the problem of learning. *Hippocampus*, 18(12):1283–1300.

Wilming, N., P. König, S. König, and E. A. Buffalo
2018. Entorhinal cortex receptive fields are modulated by spatial attention, even without movement. *eLife*, 7.

Yartsev, M. M. and N. Ulanovsky
2013. Representation of three-dimensional space in the hippocampus of flying bats. *Science*, 340(6130):367–372.

Yartsev, M. M., M. P. Witter, and N. Ulanovsky
2011. Grid cells without theta oscillations in the entorhinal cortex of bats. *Nature*, 479:103 EP –.

Yoon, K., M. A. Buice, C. Barry, R. Hayman, N. Burgess, and I. R. Fiete
2013. Specific evidence of low-dimensional continuous attractor dynamics in grid cells. *Nature Neuroscience*, 16:1077–1084.

Yoon, K., S. Lewallen, A. Kinkhabwala, T. D., and I. Fiete
2016. Grid cell responses in 1d environments assessed as slices through a 2d lattice. *Neuron*, 89:1086–99.

Appendices

A   Rank of random projection is maximal

In the paper we state that the random projection $A \in \mathbb{R}^{2M \times N}$ has rank N with "high" probability if $N \leq 2M$. We will show below that the set of $(2M \times N)$-matrices of rank less than N can be expressed as the zero set of a polynomial (which is not the zero polynomial). Thus its measure is zero and hence the probability of sampling one of these lower-rank matrices is zero as well. One way to construct such a polynomial is as follows. For $I = (i_1, \ldots, i_N)$ let A_I denote the $(N \times N)$-matrix formed by the rows of A whose indices are listed in I, i.e. the entry in row k and column l of A_I is given by $a_{i_k,l}$. Recall that for any square matrix we can compute its determinant, and recall further that the determinant is a polynomial expression. With this in hand we can now define a polynomial function f on the set of $(2M \times N)$-matrices whose zero set is given by matrices of rank less than N:

$$f(A) := \sum_{1 \leq i_1 < \ldots < i_N \leq N} \det(A_I)^2.$$

Note that $f(A)$ is non-zero, if and only if at least one of the summands is non-zero. The latter is equivalent to N of the $2M$ rows of A being linearly independent, which means that the rank is N.

B   Existence of preferred directions

For any projection matrix $A \in \mathbb{R}^{2 \times N}$ of maximum rank N there exist two *preferred* non-null directions such that the responses look perfectly triangular. This translates to the problem of finding a pair of vectors $v, w \in \mathbb{R}^N$ that satisfies the following properties: (i) They are orthogonal, i.e. $\langle v, w \rangle = 0$; (ii) They map to the standard basis in $\mathbb{R}^2$, i.e. $A(v) = e_1$ and $A(w) = e_2$; (iii) Their length is equal, i.e. $\|v\|^2 - \|w\|^2 = 0$. We can already see that we can expect such a pair to exist. Note that all three properties amount to 6 equations and our hypothesis space is $2N$-dimensional since it is the product of two N-dimensional Euclidean spaces. Thus for $N > 2$ we have just enough degrees of freedom to expect a solution to exist. For $N = 3$ let V and W denote the solutions to the equations in (iii). For $N > 3$ we can choose 1-dimensional, parallel, affine-linear sub-spaces of

the solutions to equation in (iii). For $v \in V$ let $w(v)$ denote the intersection of the orthogonal complement of v and W. This map is well-defined if v is not orthogonal to $\ker A$. So far the first two of the three properties are already satisfied. If we let $\|v\|$ go to infinity, $\|w(v)\|$ will approach a minimum, and in turn approaching the the orthogonal complement of $\ker A$ with v will let $\|w(v)\|$ approach infinity. Thus following a mean value theorem argument one concludes that there is a v with $\|v\| = \|w(v)\|$.

C   Details of the capacity algorithm

We determine the side length w by first determining the *resolution* of the set of modules (discussed later) and confirming that it is less than 0.5 units in each dimension. We then started with a small N-dimensional box of radius 0.5, and we incrementally expand the box outward and check whether the frontier region contains any collisions, splitting this frontier into a set of N-dimensional boxes and thoroughly checking each of these boxes for the starting point's representation. There's no analytic way to detect whether a particular set of module phases occur together within a N-dimensional box, but in some cases it's trivial. For example, if one of the phases never occurs anywhere in the box (while accounting for Δ), then it's clear that the set of phases never co-occur in the box. We combine this heuristic with a divide-and-conquer algorithm to split the box into smaller boxes in which at least one phase never occurs. For each box we test a few points in the box to see if they have this representation, then we test whether the box excludes any of the individual phases, and if both checks fail we split the box in half and try again on each half. If the representation is not present, this process will successfully partition the box into smaller boxes that each have this property. If the representation does occur, this process will narrow in on a range of points with representations similar to this representation until it finds it. In this way, we thoroughly search the N-dimensional volume for collisions without needing to choose a fixed high sampling density.

In each module we anchor $\phi_0 = \vec{0}$ to location $\vec{x}_0 = \vec{0}$, so A assigns phase $\vec{\phi}$ to a location $\vec{x}$ via $A\vec{x} \mod 1$. In the algorithm above we need to check whether various N-dimensional boxes contain any points near phase $\vec{0}$. To perform this check in a way that properly considers distances between phases, we decompose A into two matrices: P, a linear mapping from $\mathbb{R}^N$ to the plane $\mathbb{R}^2$, and L which specifies the lattice on that plane, such that $A = L^{-1}P$. We always set L to create a hexagonal lattice, i.e.

$$L = \begin{bmatrix} \cos 0° & \cos 60° \\ \sin 0° & \sin 60° \end{bmatrix}. \tag{7}$$

The matrix P maps a location to a point on the plane that contains the hexagonal lattice:

$$P = \frac{1}{\lambda} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \hat{v}_1 & \hat{v}_2 & \cdots & \hat{v}_N \end{bmatrix}^{-1}. \tag{8}$$

Here $\hat{v}_1$ and $\hat{v}_2$ are orthogonal N-dimensional unit vectors that define this plane, and $\hat{v}_3, \ldots, \hat{v}_N$ define the kernel of P. The period or scale of the grid is defined by λ. The distance between two phases is equal to the shortest distance between the phases on this plane.

To determine whether an N-dimensional box contains a point with a phase near $\vec{0}$, we apply transformation P to each corner of the box to obtain its shadow on the plane. We enumerate nearby lattice points on the plane, drawing circles around each lattice point with diameter equal to the phase resolution Δ, and then we check whether any of these circles intersect the shadow or are contained within the shadow. We combine this technique with the above "divide-and-conquer" and "expanding box" techniques to determine the range over which the code is unique.

We defined the *resolution* of a set of modules as the distance one must move along each dimension before the representation becomes distinguishable from the starting representation. It's possible for this resolution to be more precise in some dimensions than others, for example if all of the modules have similar projection planes and similar kernels the resolution on the projection plane will be precise but along the kernel it will be more coarse. To obtain a single number characterizing this resolution, we numerically computed the smallest N-dimensional hypercube centered at the origin for which every point on the hypercube's surface was distinguishable from the origin. We checked each face of the hypercube using the same divide-and-conquer strategy as above. The resolution is equal to half the side-length of this smallest hypercube, and we computed it to a precision of 0.01 units.
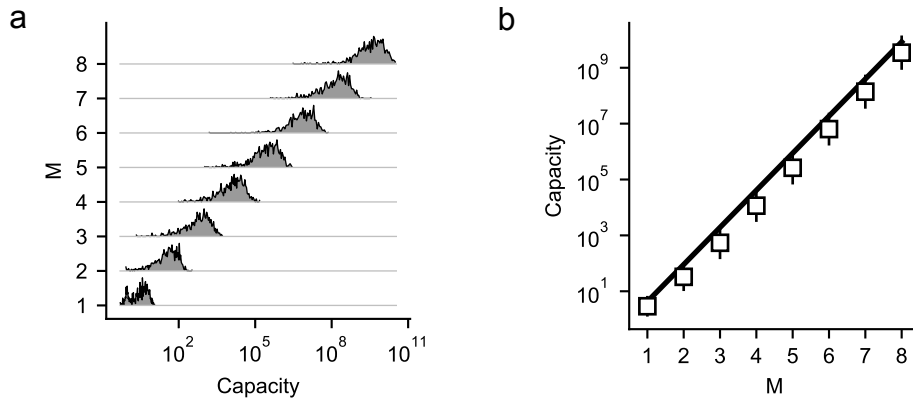
Figure 7: **1-dimensional capacity.** **(a)** Histograms of the 1D capacity data. Note that for each M the distribution is roughly log-normally distributed (1000 data-points for each M). For all computations the phase resolution is $\Delta = 0.2$. **(b)** The 1D capacity of our randomized approach (blue error-bars). We show geometric mean and standard deviation of the data (1000 data-points for each M). The capacity grows proportional to the benchmark $\Delta \cdot (1/\Delta^2)^{0.95 \cdot M}$ (thick black line); cf. (Fiete et al., 2008).
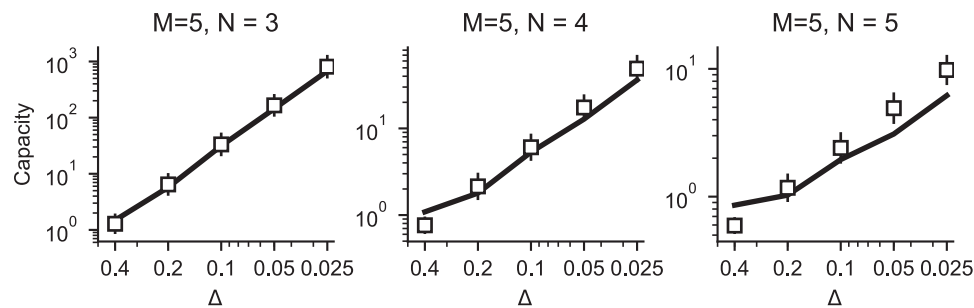

Figure 8: **Capacity as a function of phase resolution.** Capacity grows as a power of phase resolution, regardless of the dimensionality of the encoded variable
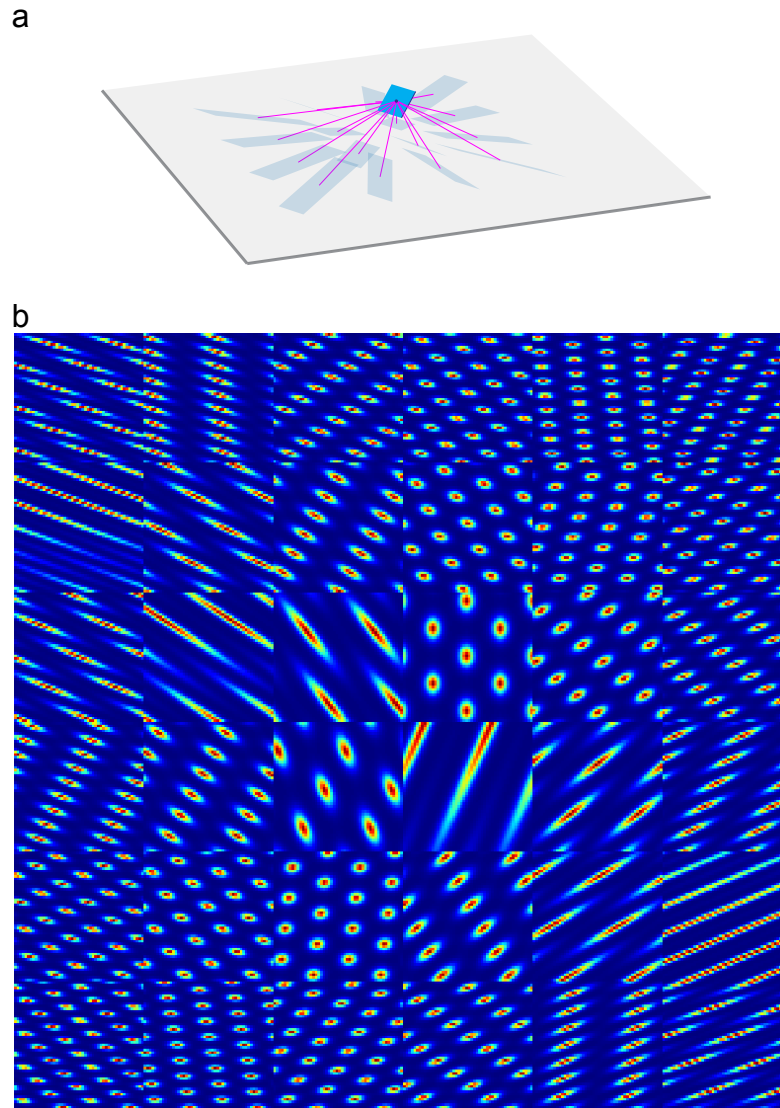
16

Figure 9: **A zoo of firing fields. (a)** A tilted plane (blue) in 3D space and 16 different projections (magenta lines) onto a common 2D input subspace (gray) along which the responses are an equilateral triangular lattice. **(b)** 6×6 different Firing fields on the blue plane induced by a family of 6×6 projections whose angles vary as indicated in (a).

17