Universal conditions for exact path integration in neural systems

John B. Issa[1] and Kechen Zhang[1]

Department of Biomedical Engineering, The Johns Hopkins University School of Medicine, Baltimore, MD 21205

Edited by Charles F. Stevens, The Salk Institute for Biological Studies, La Jolla, CA, and approved March 15, 2012 (received for review December 5, 2011)

Animals are capable of navigation even in the absence of prominent landmark cues. This behavioral demonstration of path integration is supported by the discovery of place cells and other neurons that show path-invariant response properties even in the dark. That is, under suitable conditions, the activity of these neurons depends primarily on the spatial location of the animal regardless of which trajectory it followed to reach that position. Although many models of path integration have been proposed, no known single theoretical framework can formally accommodate their diverse computational mechanisms. Here we derive a set of necessary and sufficient conditions for a general class of systems that performs exact path integration. These conditions include multiplicative modulation by velocity inputs and a path-invariance condition that limits the structure of connections in the underlying neural network. In particular, for a linear system to satisfy the path-invariance condition, the effective synaptic weight matrices under different velocities must commute. Our theory subsumes several existing exact path integration models as special cases. We use entorhinal grid cells as an example to demonstrate that our framework can provide useful guidance for finding unexpected solutions to the path integration problem. This framework may help constrain future experimental and modeling studies pertaining to a broad class of neural integration systems.

commutativity | attractor network | oscillatory interference | dead reckoning | Fourier analysis

Even without allothetic or environmental cues, animals are capable of finding their way home (1, 2), a process known as path integration or dead reckoning. This "integrative process" (3), whereby an internal representation of position is updated by incoming inertial or self-motion cues, was hypothesized over a century ago by Darwin (4) and Murphy (3). More recently, potential neural correlates of path integration have been discovered. For example, cells in brain regions associated with the Papez circuit can signal the heading direction of an animal (5, 6); grid cells in the entorhinal cortex presumably integrate this information and other self-motion cues to form a periodic spatial code (7–9); and, downstream in the hippocampus, place cells exhibit a sparser location code (10, 11).

Many computational models of path integration have been proposed. For instance, two leading classes of models of grid cells are continuous attractor networks (12–14) and oscillatory interference models (15–20). These diverse models seemingly describe a diverse class of systems; however, deeper computational principles may exist that unify the different cases of neural integration. Here we attempt to identify a general principle of path integration by starting with the exact requirement of invariance to movement trajectory in an arbitrary number of dimensions. This general approach allows us to derive a set of necessary and sufficient conditions for path integration and, for linear systems, to find explicit solutions. This framework can help unify existing models of path integration and also guide the search for unexplored solutions. We demonstrate this utility by modeling various path integration systems such as grid cells, and we show that several existing path integration models adhere to our framework.

Model

General System. We start with the most general case by considering movement in a D-dimensional space with location coordinates described by $\mathbf{x} = (x_1, \cdots, x_D)^{\mathrm{T}}$ and movement velocity given by $\mathbf{v} = (v_1, \cdots, v_D)^{\mathrm{T}} = d\mathbf{x}/dt$, where T indicates transpose. For example, we have $D = 2$ for location of a rat on the floor of a room. We use an N-dimensional column vector $\mathbf{u}$ to describe the state or activity of a network with N neurons. We assume that the network obeys a generic dynamical equation of the form $d\mathbf{u}/dt = \mathbf{h}(\mathbf{u}, \mathbf{v})$, where the function $\mathbf{h}$ does not depend explicitly on location $\mathbf{x}$ because here we focus only on velocity input and ignore any landmark-based cues. As the location $\mathbf{x}$ changes in time, the activity $\mathbf{u}$ can be solved as a function of time from the dynamical equation. We say that the system performs exact *path integration* if, starting from fixed initial conditions, the activity $\mathbf{u}$ at a final location depends only on that location, irrespective of the movement trajectory leading to that position (Fig. 1*A*). In other words, now the activity $\mathbf{u}$ is *path invariant* and can be regarded as an implicit function of location $\mathbf{x}$. The path invariance implies that $d\mathbf{u}/dt = \mathbf{h}(\mathbf{u}, \mathbf{v}) = \mathbf{F}(\mathbf{u})\mathbf{v}$ for some function $\mathbf{F}$. That is, the dynamical equation can be written as

$$\frac{d\mathbf{u}}{dt} = \mathbf{F}(\mathbf{u})\mathbf{v}, \qquad [1]$$

where matrix $\mathbf{F}$ depends on the network state $\mathbf{u}$ but not on the velocity $\mathbf{v}$. To see this, consider the derivative chain rule, $d\mathbf{u}/dt = (\partial\mathbf{u}/\partial\mathbf{x})(d\mathbf{x}/dt) = (\nabla\mathbf{u})\mathbf{v}$, where the Jacobian matrix $\nabla\mathbf{u} = \partial\mathbf{u}/\partial\mathbf{x}$ depends implicitly on $\mathbf{x}$ because $\mathbf{u}$ is an implicit function of $\mathbf{x}$. Assuming this implicit function is locally invertible, we may write $\nabla\mathbf{u} = \mathbf{F}(\mathbf{u})$ and obtain Eq. **1**.

Exact path integration by Eq. **1** requires an additional condition. For simplicity we show the argument for $D = 2$ here (for generalization to higher dimensions, see *SI Text 1*). Consider the differential form $d\mathbf{u} = \mathbf{f}^1 dx_1 + \mathbf{f}^2 dx_2$ of Eq. **1**, where $\mathbf{f}^k$ is the kth column of $\mathbf{F}$, which depends implicitly on $\mathbf{x}$. If activity $\mathbf{u}$ is path invariant, its cumulative increment over any closed movement trajectory C must vanish: $\oint_C d\mathbf{u} = \mathbf{0}$. By Green's theorem, this means that $\oint_C d\mathbf{u} = \iint_R (\partial\mathbf{f}^2/\partial x_1 - \partial\mathbf{f}^1/\partial x_2) dx_1 dx_2 = \mathbf{0}$ for any region R bounded by C. Hence $\partial\mathbf{f}^2/dx_1 = \partial\mathbf{f}^1/dx_2$. For arbitrary dimension D, this condition becomes

$$\frac{\partial f_i^j}{\partial x_k} = \frac{\partial f_i^k}{\partial x_j}, \qquad [2]$$

where f_i^j is the entry in the ith row and jth column of matrix $\mathbf{F}$ ($i = 1, \cdots, N$ and $j, k = 1, \cdots, D$ for $D \geq 2$). Eq. **2** specifies a

Author contributions: J.B.I. and K.Z. designed research, performed research, analyzed data, and wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. E-mail: john.issa@gmail.com or kzhang4@jhmi.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1119880109/-/DCSupplemental.

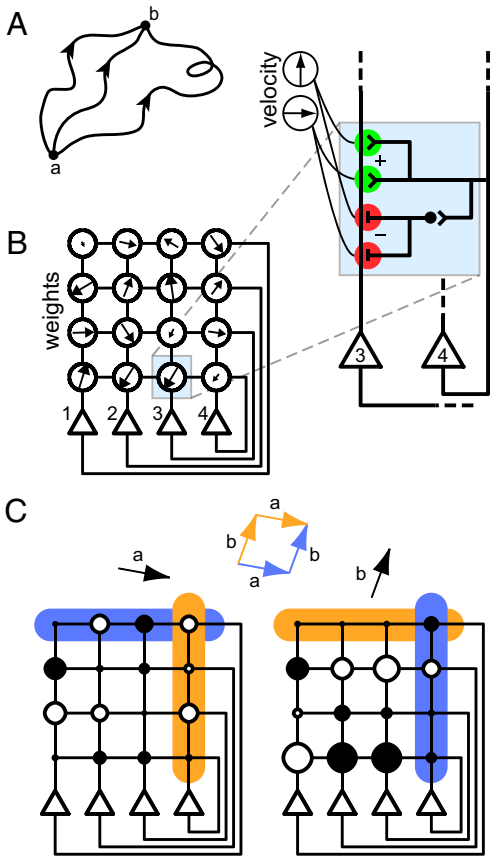

Fig. 1. Constraints on a path integration network. (*A*) For the system to perform exact path integration, the network state at location *b* should be the same regardless of which path it took to get there from location *a*. (*B*) A four-unit network with each recurrent connection node characterized by a weight vector ($\mathbf{w}_{ij}$ in Eq. **4**), shown as an arrow that represents the preferred direction and the magnitude of velocity modulation. The effective weight of each connection is the dot product between this weight vector $\mathbf{w}_{ij}$ and the movement velocity **v**. The connection from neuron 4 to neuron 3 (*Left*, shaded box) is expanded (*Right*), showing that each weight vector can be implemented by summing excitatory synapses (green) and inhibitory synapses (red, via interneurons) if these synapses are multiplicatively modulated by the component velocity inputs. (*C*) Illustration of commutativity condition (Eq. **5**) for a linear network when one of two paths is followed: **ab** (blue) or **ba** (orange). For movement in a straight line (**a** or **b**), the effective weights are represented by the size of the dots and whether they are open (positive value) or closed (negative value). For the commutativity condition to hold, the dot product of the blue vectors should equal the dot product of the orange vectors. The blue vector (shaded) on the left represents the outputs from neuron 1 for direction **a**, whereas the blue vector on the right represents the inputs to neuron 4 for direction **b** (and vice versa for the orange vectors).

necessary and sufficient condition that guarantees that the differential form $d\mathbf{u} = \mathbf{F}d\mathbf{x}$ is integrable with respect to the spatial variable **x**, which in turn means that activity depends on location only and not on the movement trajectory. For $D = 2$ or 3, Eq. **2** is equivalent to the curl-free condition for a gradient vector field ($\mathbf{F} = \nabla\mathbf{u}$). In arbitrary dimensions, the general case of Eq. **2** follows from the Poincaré lemma (*SI Text 1*).

Eq. **1** implies that the rate of state change should be modulated linearly by the velocity **v**. As shown in Fig. 1*B*, one way to implement this general requirement is a model system where recurrent synapses are multiplicatively modulated by velocity inputs (see *Discussion* for other possibilities). Finally, note that Eq. **1** (as well as Eq. **3** below) does not contain a decay term such as $-\mathbf{u}$ commonly seen in neural models. We explain how this may be justified in *SI Text 6*.

Linear System. To gain a better understanding of the dynamical equation (Eq. **1**) and the path-invariance condition (Eq. **2**), we simplify the system by assuming that matrix $\mathbf{F}(\mathbf{u})$ is a linear function of state **u**. That is, for $\mathbf{F}(\mathbf{u}) = [\mathbf{f}^1(\mathbf{u}), \cdots, \mathbf{f}^D(\mathbf{u})]$, we assume each column is a vector given by $\mathbf{f}^k(\mathbf{u}) = [f_1^k(\mathbf{u}), \cdots, f_N^k(\mathbf{u})]^{\mathrm{T}} = \mathbf{W}^k\mathbf{u}$, with $\mathbf{W}^k = \{w_{ij}^k\}_{i,j=1,\cdots,N}$ being a weight matrix for each spatial dimension $k = 1, \cdots, D$. Here w_{ij}^k is the weight of the connection from unit j to unit i, and the superscript k indicates the spatial dimension of velocity modulation. Now we can reformulate Eq. **1** as

$$\frac{d\mathbf{u}}{dt} = \sum_{k=1}^{D} \mathbf{W}^k\mathbf{u}v_k = \left(\sum_{k=1}^{D} \mathbf{W}^k v_k\right)\mathbf{u} = \mathbf{W}^{\mathrm{eff}}\mathbf{u}, \qquad [3]$$

where $\mathbf{W}^{\mathrm{eff}} = \{w_{ij}^{\mathrm{eff}}\}_{i,j=1,\cdots,N} = \sum_{k=1}^{D} \mathbf{W}^k v_k$ is defined as the effective weight matrix. The effective weight from unit j to unit i can be rewritten as:

$$w_{ij}^{\mathrm{eff}} = \sum_{k=1}^{D} w_{ij}^k v_k = \mathbf{w}_{ij} \cdot \mathbf{v}, \qquad [4]$$

which is a dot product between the instantaneous velocity **v** and a fixed weight vector $\mathbf{w}_{ij} = (w_{ij}^1, \cdots, w_{ij}^D)^{\mathrm{T}}$. Thus, the effective weight is largest when velocity is aligned with the direction of the weight vector, zero when the two vectors are orthogonal, and negative when the angle between the two vectors is >90°. As shown in Fig. 1*B*, the velocity-modulated connection between a pair of units can be characterized by a single vector with the appropriate orientation and magnitude.

Working with a linear system allows us to simplify the condition from Eq. **2** to the following:

$$\mathbf{W}^j\mathbf{W}^k = \mathbf{W}^k\mathbf{W}^j \qquad [5]$$

for all $j, k = 1, \cdots, D$ (*SI Text 1*). In other words, the weight matrices for different spatial dimensions must commute.

Results

Eigensolutions. We examine how the linear dynamical system in Eq. **3** generates activity **u** as a function of location **x**. For all of the examples considered in this paper, with the exception of the linear tuning function (see Fig. 4*A*), we assume that the weight matrices $\mathbf{W}^1, \cdots, \mathbf{W}^D$ have distinct eigenvalues and are thus diagonalizable (general solutions without this assumption are shown in *SI Text 2*). Because diagonalizable matrices share the same eigenvectors if and only if they commute, the commutative condition in Eq. **5** implies that these matrices have an identical set of eigenvectors, although their eigenvalues may differ.

Whereas the solution to Eq. **3** specifies the activity **u** as a function of time, we can replace time t with the position $\mathbf{x}(t)$ for any trajectory because of path invariance. For simplicity we choose our trajectory to start at the origin **0** and move toward **x** at a constant velocity. Now Eq. **3** becomes a homogenous linear differential equation with constant coefficients and its solution is a linear combination of basis functions,

$$\mathbf{u}(\mathbf{x}) = \sum_{i=1}^{N} c_i\mathbf{z}_i \exp\left(\mathbf{p}_i^{\mathrm{T}}\mathbf{x}\right), \qquad [6]$$

where $\mathbf{p}_i = (\lambda_i^1, \cdots, \lambda_i^D)^{\mathrm{T}}$, λ_i^k is the ith eigenvalue of matrix $\mathbf{W}^k$, $\mathbf{z}_i$ is the associated eigenvector, and coefficient $c_i = \mathbf{z}_i^{\mathrm{T}}\mathbf{u}(\mathbf{0})$. With properly chosen eigenvectors and eigenvalues, this basis function set $\{\mathbf{z}_i \exp(\mathbf{p}_i^{\mathrm{T}}\mathbf{x})\}$ can, in principle, approximate any smooth function to arbitrary precision because it can at least provide Fourier components when the eigenvalues are imaginary (*SI Text*

2). Weights for the linear system can be learned by a local supervised learning rule (*SI Text 4* and Fig. S1).

Illustrating the Commutative Condition. At first glance, the commutative condition (Eq. **5**) is not intuitive in a biological context. For a more concrete illustration, we restate the condition as a discrete version for arbitrary movement directions. First, we choose two displacement vectors **a** and **b** (Fig. 1*C*). Using the matrix exponential to solve Eq. **3**, we have $\mathbf{u}(\mathbf{a}) = \mathbf{W^a}\mathbf{u}(\mathbf{0})$, where $\mathbf{W^a} = \exp(\sum_{j=1}^{D} \mathbf{W}^j a_j)$ with $\mathbf{a} = (a_1, \cdots, a_D)^{\mathrm{T}}$. It is easy to see that path invariance requires commutativity of $\mathbf{W^a}$ with $\mathbf{W^b}$ because the final activity along path **ab** is $\mathbf{u}(\mathbf{ab}) = \mathbf{W^b}\mathbf{W^a}\mathbf{u}(\mathbf{0})$, which should equal $\mathbf{u}(\mathbf{ba})$, implying $\mathbf{W^b}\mathbf{W^a} = \mathbf{W^a}\mathbf{W^b}$ for all $\mathbf{a}, \mathbf{b}$, including movements in cardinal directions (as in Eq. **5**).

Now we look at any pair of units and define the net influence of unit i on unit j along path **ab** as the dot product of two sets of connections for two movement segments: those from unit i to all units for movement **a** (ith row of $\mathbf{W^a}$ and blue row in Fig. 1*C*) and those from all units to unit j for movement **b** (jth row of $\mathbf{W^b}$ and blue column in Fig. 1*C*). By Eq. **5**, this net influence must stay the same along path **ba**. Thus, for a pair of units, path invariance puts restriction on connections involving all units in the network and not just on direct connections.

Grid Cell Simulation. Grid cells in the rat medial entorhinal cortex fire in a hexagonal pattern. Nearby cells have the same orientation and spatial frequency but differing spatial phases, and their firing patterns are maintained in the dark (7). Here we simulate the spatial pattern of grid cell firing with a linear system governed by Eqs. **3** and **5**. We show that this method can guide us to find diverse solutions that generate the same hexagonal activity pattern.

All of the examples in this section are based on sinusoidal basis functions, which are special cases of the exponential functions in Eq. **6** with purely imaginary eigenvalues (*SI Text 2*). We first consider a minimal network. Because a hexagonal grid can be generated by combining three sinusoids, the smallest network possible has six neurons. The connection scheme (Fig. 2*A*) does not show any obvious hexagonal symmetry and may even look quite random, yet the weight matrices commute and can generate a hexagonal activity pattern by integrating velocity inputs (Fig. 2 *B* and *C*). Each cell has the same activity pattern but spatially shifted according to their relative spatial phases.

As another example, we choose a regular arrangement of spatial phases and a regular connection pattern where each neuron receives symmetric inputs from its six nearest neighbors (Fig. 3 and Fig. S2*A*). In this example, the hexagonal activity pattern arises due to the inherent hexagonal symmetry of the connections. Each neuron connects to a uniformly spaced ring of other neurons and each weight vector is proportional to the relative spatial positions of the connected neurons, ensuring that the weight matrices commute (Fig. S2 and *SI Text 3*).

We also studied a large network with many neurons with random spatial phases and found that, on average, there was a weak preference for preferred velocities out of phase with the grid orientation (Fig. S3 *B–D*). By contrast, for the case of evenly spaced spatial phases as in Fig. 3, the preferred weight directions all lined up exactly with the grid (Fig. S3*A*), an observation that is more consistent with grid cell studies (21), hinting that such biological systems may favor a radially symmetric connection scheme.

Example Solutions from the Linear System. Any linear path integration system can be built using the eigensolutions described above after identifying the dimensionality of the system and the basis functions of the desired spatial activity pattern. A broad class of path integration systems can be simulated in this way, covering various dimensionalities and types of space, either periodic or nonperiodic. Several examples of these linear systems are shown in Fig. 4. In the one-dimensional examples, a sinusoid solution

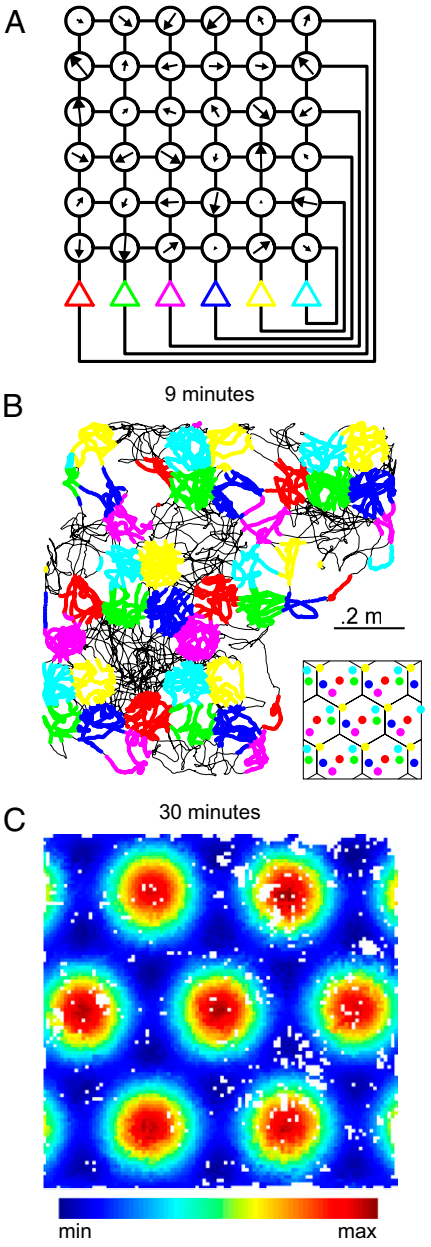

Fig. 2. A six-unit grid cell network. (*A*) Weights were found for a network with randomly chosen spatial phases. This network is shown using the same convention as in Fig. 1*B*. (*B*) Simulated trajectory over 9 min is shown (black line). To display activity, spikes are plotted as colored dots whenever a unit's activity crosses 75% of its maximum. If multiple units simultaneously cross the threshold, only the unit with highest activity is displayed. Color scheme is same as in *A*. *Inset* shows the spatial phases for each unit on top of the underlying hexagonal tiles. (*C*) For a full 30-min session, the average unit activity at each position (100 × 100 bins) for the first neuron (previously in red) is shown. White indicates positions not visited in the simulation. See *SI Text 7* for further details.

(Fig. 4*A*, *Left*) is generated by using a weight matrix with a pair of imaginary eigenvalues, whereas a linear activity pattern (Fig. 4*A*, *Right*) is generated by using a weight matrix with repeated eigenvalues of zero. In higher dimensions, arbitrarily complex patterns can be generated, ranging from an image of a human eye to a hypothetical 3D grid cell (Fig. 4 *B* and *C*). Because the solution set scales with the number of neurons and not dimensionality, a minimal model of 3D grid cells requires only two additional neurons beyond the six-unit minimal model of 2D grid cells.

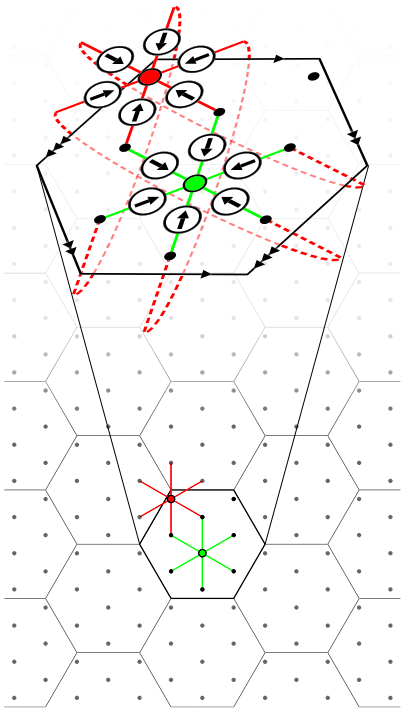

Fig. 3. A nine-unit grid cell network with spatial symmetry. Each unit connects to its six nearest neighbors. These connections are illustrated for two of the units (green and red dots). Arrows indicate preferred directions for each weight. For the red unit, the connections wrap around the tile to the opposite edge (identified by the arrowheads). A tessellation of the hexagonal tile is also shown, illustrating that these units are evenly spaced.

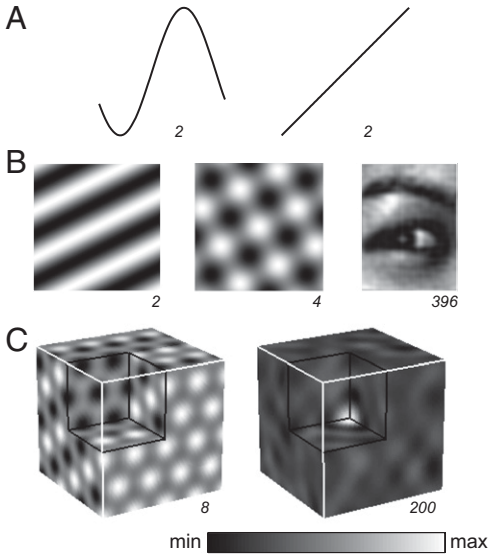

Fig. 4. Diverse activity patterns can be generated by a linear network performing exact path integration in multiple dimensions. (*A*) In one dimension, the solution set includes sinusoids (*Left*, periodic as seen in head direction cells) and first-order polynomials (*Right*, nonperiodic as seen in the eye position integrator of the oculomotor system). (*B*) In two dimensions, the solutions are superpositions of one-dimensional patterns. These patterns can be a single sinusoidal grating (*Left*), two gratings at 90° to form a square grid (*Center*; this pattern is generated by the four-unit network of Fig. 1*B*), or a large number of sinusoids to form a representation of an image of a human eye (*Right*). (*C*) These solutions generalize to higher dimensions. In three dimensions, a grid based on the hexagonal close-packed lattice (*Left*) or a place cell based on a large number of random-frequency zero-phase sinusoids (*Right*) can be formed. The numbers below indicate the number of units used to generate that spatial pattern.

Existing Exact Path Integration Models. We chose four disparate models of exact path integrators and asked whether they adhere to our conditions. These four models are a linear eye position integrator, nonlinear bump attractor networks, oscillatory interference models, and a 3D head orientation model. Although the actual computational mechanisms in these examples are radically different, we find that these models all obey the same mathematical conditions of Eqs. **1** and **2** to achieve exact path integration (see *SI Text 5* for details and proofs). Briefly, a linear eye position integrator model that produces firing rates proportional to position (22, 23) is consistent with our theory in one-dimensional space. The continuous attractor network (24–26) supports a stable activity pattern that can be moved by velocity inputs. We must first let the system settle to a stable activity pattern, after which it performs exact path integration. The oscillatory interference models (15, 16, 20, 27) of grid cells and place cells perform path integration by modulating the oscillator frequencies. Although the activity of the interfering oscillators is not spatially invariant, the envelope is. Finally, in a recent vestibular model (28), angular velocity, an inexact differential quantity, is integrated to calculate head orientation. We must first convert to a coordinate system where velocity is an exact differential quantity. With respect to the new coordinates, the system is an exact integrator.

Discussion

Path integration is a general mechanism used in various neural models. Insight into the fundamental computational principles behind accurate path integration can both unify existing models and provide guidance for future investigations into unexplored mechanisms. Here we have derived necessary and sufficient conditions for exact path integration under very general assumptions. These two conditions, multiplicative modulation and equivalence of mixed partial derivatives, arise from the chain rule and the Poincaré lemma, respectively. Each of these results bears theoretical and experimental implications for path integration.

The first condition, multiplicative modulation of inputs by velocity, is intuitively sound because doubling the speed should double the drive to move the system along the manifold of position. Various neural systems have been shown to use multiplication, including gain fields of parietal area 7a neurons (29), looming-sensitive neurons of the locust visual system (30), and conjunctive cells of the entorhinal cortex (31). Various models exist to explain how this multiplication could be implemented in a biologically plausible manner. Exponentiation by active membrane conductances of summed logarithmic synaptic inputs gives rise to an effective multiplication of inputs (32, 33). Another dendritic mechanism is that of nonlinear interaction of nearby inputs on a dendritic tree to yield multiplication followed by summation at the soma of more distal inputs (34). Alternatively, multiplication can be achieved through network effects via a properly tuned recurrent network (35). Finally, there are hints that synchrony of inputs and interplay with the theta rhythm could allow for one input to modulate other inputs (36, 37). Multiplication in a real biological system may require additional mechanisms to deal with negative weights and the degenerate case of zero net input at zero velocity.

The second condition constrains the connectivity of a path-invariant network. Computationally, equality of mixed partial derivatives provides a robust condition for identifying whether a model is path invariant. Experimentally, verifying path invariance and linking connectivity to activity patterns are more difficult. One way to test the equality of mixed partial derivatives is by in vivo patch clamp of a neuron to measure synaptic currents over a variety of paths (38). Whereas these data could

provide support for path invariance, a complete network needs to be identified to have a full characterization of the dynamics of the system. For example, in our small grid cell network in Fig. 2, even if it could be measured experimentally, the connection matrix exhibits no clear symmetry, yet it produces hexagonal symmetry in its activity. Thus, one needs an explicit, quantitative model for such cases. Recent technological advancements would allow a model to be tested against simultaneously measured connectivity and activity. By combining focal optogenetic control of neurons (39), two-photon calcium imaging of a large number of neurons (40), recordings from freely moving animals via virtual reality (41) or miniaturization of optics (42), and reconstruction of neural connectivity via serial EM (43) or postmortem in vitro path clamping (44), one could both record and manipulate the functional and anatomical properties of path-invariant neurons on a large scale. Even then, accurate measurement of all network parameters would be difficult but simplifications, such as a linear approximation studied here, could make the problem more tractable.

Finally, although we have focused on spatial invariance in this paper, our results are not just limited to systems that are involved in navigation. The many other neural systems that use integration (45, 46) conform to our results as well, provided that they can be cast under our mathematical framework.

ACKNOWLEDGMENTS. We thank C. DiMattina, M. B. Johny, J. D. Monaco, H. T. Blair, and J. J. Knierim for discussions and two anonymous reviewers for suggestions. This work was supported by National Institutes of Health Grant R01 MH079511.

1. Müller M, Wehner R (1988) Path integration in desert ants, Cataglyphis fortis. *Proc Natl Acad Sci USA* 85:5287–5290.
2. Mittelstaedt M-L, Mittelstaedt H (1980) Homing by path integration in a mammal. *Naturwissenschaften* 67:566–567.
3. Murphy JJ (1873) Instinct: A mechanical analogy. *Nature* 7:483.
4. Darwin C (1873) Origin of certain instincts. *Nature* 7:417–418.
5. Taube JS (2007) The head direction signal: Origins and sensory-motor integration. *Annu Rev Neurosci* 30:181–207.
6. Wiener SI, Taube JS (2005) *Head Direction Cells and the Neural Mechanisms of Spatial Orientation* (MIT Press, Cambridge, MA).
7. Hafting T, Fyhn M, Molden S, Moser M-B, Moser EI (2005) Microstructure of a spatial map in the entorhinal cortex. *Nature* 436:801–806.
8. Moser EI, Kropff E, Moser M-B (2008) Place cells, grid cells, and the brain's spatial representation system. *Annu Rev Neurosci* 31:69–89.
9. Giocomo LM, Moser M-B, Moser EI (2011) Computational models of grid cells. *Neuron* 71:589–603.
10. O'Keefe J, Dostrovsky J (1971) The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Res* 34:171–175.
11. Wilson MA, McNaughton BL (1993) Dynamics of the hippocampal ensemble code for space. *Science* 261:1055–1058.
12. Fuhs MC, Touretzky DS (2006) A spin glass model of path integration in rat medial entorhinal cortex. *J Neurosci* 26:4266–4276.
13. Burak Y, Fiete IR (2009) Accurate path integration in continuous attractor network models of grid cells. *PLoS Comput Biol*, 10.1371/journal.pcbi.1000291.
14. Navratilova Z, Giocomo LM, Fellous J-M, Hasselmo ME, McNaughton BL (2011) Phase precession and variable spatial scaling in a periodic attractor map model of medial entorhinal grid cells with realistic after-spike dynamics. *Hippocampus* 22:772–789.
15. Burgess N, Barry C, O'Keefe J (2007) An oscillatory interference model of grid cell firing. *Hippocampus* 17:801–812.
16. Hasselmo ME, Giocomo LM, Zilli EA (2007) Grid cell firing may arise from interference of theta frequency membrane potential oscillations in single neurons. *Hippocampus* 17:1252–1271.
17. Blair HT, Gupta K, Zhang K (2008) Conversion of a phase- to a rate-coded position signal by a three-stage model of theta cells, grid cells, and place cells. *Hippocampus* 18:1239–1255.
18. Jeewajee A, Barry C, O'Keefe J, Burgess N (2008) Grid cells and theta as oscillatory interference: Electrophysiological data from freely moving rats. *Hippocampus* 18: 1175–1185.
19. Burgess N (2008) Grid cells and theta as oscillatory interference: Theory and predictions. *Hippocampus* 18:1157–1174.
20. Monaco JD, Knierim JJ, Zhang K (2011) Sensory feedback, error correction, and remapping in a multiple oscillator model of place-cell activity. *Front Comput Neurosci*, 10.3389/fncom.2011.00039.
21. Doeller CF, Barry C, Burgess N (2010) Evidence for grid cells in a human memory network. *Nature* 463:657–661.
22. Cannon SC, Robinson DA, Shamma S (1983) A proposed neural network for the integrator of the oculomotor system. *Biol Cybern* 49:127–136.
23. Seung HS (1996) How the brain keeps the eyes still. *Proc Natl Acad Sci USA* 93: 13339–13344.
24. Wilson HR, Cowan JD (1972) Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys J* 12:1–24.
25. McNaughton BL, Battaglia FP, Jensen O, Moser EI, Moser M-B (2006) Path integration and the neural basis of the 'cognitive map'. *Nat Rev Neurosci* 7:663–678.
26. Knierim JJ, Zhang K (2012) Attractor dynamics of spatially correlated neural activity in the limbic system. *Annu Rev Neurosci*, 10.1146/annurev-neuro-062111-150351.
27. Welday AC, Shlifer IG, Bloom ML, Zhang K, Blair HT (2011) Cosine directional tuning of theta cell burst frequencies: Evidence for spatial coding by oscillatory interference. *J Neurosci* 31:16157–16176.
28. Green AM, Angelaki DE (2007) Coordinate transformations and sensory integration in the detection of spatial orientation and self-motion: From models to experiments. *Prog Brain Res* 165:155–180.
29. Andersen RA, Bracewell RM, Barash S, Gnadt JW, Fogassi L (1990) Eye position effects on visual, memory, and saccade-related activity in areas LIP and 7a of macaque. *J Neurosci* 10:1176–1196.
30. Hatsopoulos N, Gabbiani F, Laurent G (1995) Elementary computation of object approach by wide-field visual neuron. *Science* 270:1000–1003.
31. Sargolini F, et al. (2006) Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science* 312:758–762.
32. Gabbiani F, et al. (2004) Multiplication and stimulus invariance in a looming-sensitive neuron. *J Physiol Paris* 98:19–34.
33. Rothman JS, Cathala L, Steuber V, Silver RA (2009) Synaptic depression enables neuronal gain control. *Nature* 457:1015–1018.
34. Mel BW (1993) Synaptic integration in an excitable dendritic tree. *J Neurophysiol* 70: 1086–1101.
35. Salinas E, Abbott LF (1996) A model of multiplicative neural responses in parietal cortex. *Proc Natl Acad Sci USA* 93:11956–11961.
36. Salinas E, Sejnowski TJ (2000) Impact of correlated synaptic input on output firing rate and variability in simple neuronal models. *J Neurosci* 20:6193–6209.
37. Wang H-P, Spencer D, Fellous J-M, Sejnowski TJ (2010) Synchrony of thalamocortical inputs maximizes cortical reliability. *Science* 328:106–109.
38. Harvey CD, Collman F, Dombeck DA, Tank DW (2009) Intracellular dynamics of hippocampal place cells during virtual navigation. *Nature* 461:941–946.
39. Yizhar O, Fenno LE, Davidson TJ, Mogri M, Deisseroth K (2011) Optogenetics in neural systems. *Neuron* 71:9–34.
40. Stosiek C, Garaschuk O, Holthoff K, Konnerth A (2003) In vivo two-photon calcium imaging of neuronal networks. *Proc Natl Acad Sci USA* 100:7319–7324.
41. Dombeck DA, Harvey CD, Tian L, Looger LL, Tank DW (2010) Functional imaging of hippocampal place cells at cellular resolution during virtual navigation. *Nat Neurosci* 13:1433–1440.
42. Ghosh KK, et al. (2011) Miniaturized integration of a fluorescence microscope. *Nat Methods* 8:871–878.
43. Bock DD, et al. (2011) Network anatomy and in vivo physiology of visual cortical neurons. *Nature* 471:177–182.
44. Hofer SB, et al. (2011) Differential connectivity and response dynamics of excitatory and inhibitory neurons in visual cortex. *Nat Neurosci* 14:1045–1052.
45. Robinson DA (1989) Integrating with neurons. *Annu Rev Neurosci* 12:33–45.
46. Carpenter RHS (2011) What Sherrington missed: The ubiquity of the neural integrator. *Ann N Y Acad Sci* 1233:208–213.

Supporting Information

Issa and Zhang 10.1073/pnas.1119880109

SI Text

1. Necessary and Sufficient Conditions for Path Invariance. In this section we show that a generic network described by Eq. **1** in the main text is an exact (i.e., path-invariant) integrator if and only if it satisfies Eq. **2** for partial derivatives. For a linear network, this condition reduces to the matrix commutativity condition in Eq. **5**.

As in the main text, let $\mathbf{x} = (x_1,\cdots, x_D)^{\mathrm{T}}$ be the encoded variable in D-dimensional space, $\mathbf{v} = (v_1,\cdots, v_D)^{\mathrm{T}} = d\mathbf{x}/dt$ be the velocity, and $\mathbf{u} = (u_1,\cdots, u_N)^{\mathrm{T}}$ be the activities of N neurons in a network, with T indicating transpose. Path invariance requires that the activities $\mathbf{u}$ depend implicitly on location $\mathbf{x}$, regardless of the trajectory. On the basis of the derivative chain rule

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial \mathbf{x}}\frac{d\mathbf{x}}{dt}, \qquad [S1]$$

where

$$\frac{\partial \mathbf{u}}{\partial \mathbf{x}} = \nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u_1}{\partial x_1} & \cdots & \frac{\partial u_1}{\partial x_D} \\ \vdots & \ddots & \vdots \\ \frac{\partial u_N}{\partial x_1} & \cdots & \frac{\partial u_N}{\partial x_D} \end{bmatrix} \qquad [S2]$$

is the gradient or Jacobian matrix, we argue that the dynamics of the neural network should obey Eq. **1** in the main text, which reads

$$\frac{d\mathbf{u}}{dt} = \mathbf{F}(\mathbf{u})\mathbf{v}, \qquad [S3]$$

where

$$\mathbf{F}(\mathbf{u}) = \begin{bmatrix} f_1^1(\mathbf{u}) & \cdots & f_1^D(\mathbf{u}) \\ \vdots & \ddots & \vdots \\ f_N^1(\mathbf{u}) & \cdots & f_N^D(\mathbf{u}) \end{bmatrix} = \left[\mathbf{f}^1(\mathbf{u}),\cdots,\mathbf{f}^D(\mathbf{u})\right]. \qquad [S4]$$

In general, the system in Eq. **S3** is not an exact integrator; that is, its activity $\mathbf{u}$ at a given location $\mathbf{x}$ may depend on which trajectory was actually followed to reach that location. We see intuitively by comparing Eqs. **S1** and **S3** that matrix $\mathbf{F}$ should be a gradient given by $\mathbf{F} = \nabla \mathbf{u}$ if the activity $\mathbf{u}$ obtained from Eq. **S3** is indeed path invariant.

A more precise formulation of this argument involves the Poincaré lemma, which states that a differentiable 1-form $\boldsymbol{\omega}$ defined everywhere in a D-dimensional cube is exact if and only if it is also closed (1). In our case, $\boldsymbol{\omega} = d\mathbf{u} = \mathbf{F}(\mathbf{u})d\mathbf{x} = \sum_{i=1}^{D}\mathbf{f}^i(\mathbf{u})dx_i$ with $\mathbf{u}$ depending implicitly on $\mathbf{x}$. The requirement for path invariance amounts to requiring that $\boldsymbol{\omega}$ be exact or integrable with respect to $\mathbf{x}$, meaning that there exists some function $\mathbf{g}(\mathbf{x})$ such that $\boldsymbol{\omega} = d\mathbf{g}$. For this relation to hold true a necessary and sufficient condition is that $\boldsymbol{\omega}$ must be closed; that is, its exterior derivative must vanish:

$$d\boldsymbol{\omega} = \sum_{j,k=1}^{D}\left(\frac{\partial \mathbf{f}^j}{\partial x_k} - \frac{\partial \mathbf{f}^k}{\partial x_j}\right)dx_j dx_k = \mathbf{0}. \qquad [S5]$$

It follows that mixed partial derivatives must be equal, giving us Eq. **2** from the main text:

$$\frac{\partial f_i^j}{\partial x_k} = \frac{\partial f_i^k}{\partial x_j} \qquad [S6]$$

for all $i = 1,\cdots, N$ and $j, k = 1,\cdots, D$. The argument above is valid in space of any dimension $D \geq 2$.

Next we further assume that $\mathbf{F}$ is linear, so $\mathbf{f}^k(\mathbf{u}) = [f_1^k(\mathbf{u}), f_2^k(\mathbf{u}),\cdots, f_N^k(\mathbf{u})]^{\mathrm{T}} = \mathbf{W}^k\mathbf{u}$, or $f_i^k = \sum_{j=1}^{N} w_{ij}^k u_j$, where w_{ij}^k is a weight tensor that describes the connection strength from unit j to unit i modulated by velocity in direction k. To satisfy the condition for the Poincaré lemma, $\partial \mathbf{f}^j/\partial x_k = \partial \mathbf{W}^j\mathbf{u}/\partial x_k = \mathbf{W}^j\mathbf{f}^k = \mathbf{W}^j\mathbf{W}^k\mathbf{u}$ must equal $\partial \mathbf{f}^k/\partial x_j = \mathbf{W}^k\mathbf{W}^j\mathbf{u}$ for all k, j, and $\mathbf{u}$. In the derivation above, the relation $\nabla \mathbf{u} = \mathbf{F}$ or $\partial \mathbf{u}/\partial x_k = \mathbf{f}^k$ was used. It follows that all of the weight matrices must commute; that is, for all k and j we must have

$$\mathbf{W}^j\mathbf{W}^k = \mathbf{W}^k\mathbf{W}^j. \qquad [S7]$$

This is Eq. **5** of the main text.

2. Linear System Solutions. In this section we generalize the solution to the linear system in Eq. **3** of the main text by allowing multiplicities or identical eigenvalues of the weight matrices. Then we describe a method for building a linear system with any desired activity pattern by first transforming the spatial patterns to their equivalent Fourier basis set and then calculating the weight matrices.

2.1. General form of solutions. In the main text, by assuming weight matrices with distinct eigenvalues, we found that a general solution to Eq. **3** is a linear combination of basis functions of the form $\mathbf{z}_i \exp(\mathbf{p}_i^{\mathrm{T}}\mathbf{x})$ (Eq. **6**). We now relax the restriction on distinct eigenvalues to derive a more general solution. By path invariance we know that $\mathbf{u}(t) = \mathbf{u}(\mathbf{x}(t))$, where we write $\mathbf{u}$ as a function of either time t or location $\mathbf{x}$. Given the initial condition $\mathbf{u}(\mathbf{0})$ at the origin $\mathbf{x} = \mathbf{0}$, we can calculate the activity $\mathbf{u}(\mathbf{x})$ for any position $\mathbf{x} = (x_1,\cdots, x_D)^{\mathrm{T}}$ by moving with a constant velocity $\mathbf{v} = \mathbf{x}$ so that $\mathbf{u}(t = 1) = \mathbf{u}(\mathbf{0} + \mathbf{v} \times 1) = \mathbf{u}(\mathbf{x})$. The new solution is

$$\mathbf{u}(\mathbf{x}) = \exp(\mathbf{W}^{\mathbf{x}})\mathbf{u}(\mathbf{0}) \qquad [S8]$$

with $\mathbf{W}^{\mathbf{x}} = \sum_{j=1}^{D}\mathbf{W}^j x_j$.

Using the Jordan canonical form, we can decompose each weight matrix as $\mathbf{W}^j = \mathbf{W}_{\mathrm{d}}^j + \mathbf{W}_{\mathrm{n}}^j$, where $\mathbf{W}_{\mathrm{d}}^j$ is a diagonalizable matrix and $\mathbf{W}_{\mathrm{n}}^j$ is a nilpotent matrix, with $\mathbf{W}_{\mathrm{d}}^j\mathbf{W}_{\mathrm{n}}^j = \mathbf{W}_{\mathrm{n}}^j\mathbf{W}_{\mathrm{d}}^j$ (2). Because each diagonalizable matrix $\mathbf{W}_{\mathrm{d}}^j$ can be expressed as a polynomial of the original matrix $\mathbf{W}^j$ assuming that $\mathbf{W}^j$ is nonderogatory (i.e., each distinct eigenvalue has geometric multiplicity 1, corresponding to exactly one Jordan block) (2), the commutativity of the original weight matrices in Eq. **S7** implies that the diagonalizable components of these matrices also commute; namely, $\mathbf{W}_{\mathrm{d}}^j\mathbf{W}_{\mathrm{d}}^k = \mathbf{W}_{\mathrm{d}}^k\mathbf{W}_{\mathrm{d}}^j$ for all j and k. Diagonalizable matrices that commute are simultaneously diagonalizable, so we can write $\mathbf{W}_{\mathrm{d}}^j = \mathbf{S}\mathbf{D}^j\mathbf{S}^{-1}$, where matrix $\mathbf{S}$ contains the shared eigenvectors as its columns and the diagonal matrix $\mathbf{D}^j = \mathrm{diag}(\lambda_1^j,\cdots,\lambda_N^j)$ contains all eigenvalues λ_i^j of $\mathbf{W}_{\mathrm{d}}^j$. Thus, we have

$$\mathbf{W}^{\mathbf{x}} = \mathbf{S}\left(\sum_{j=1}^{D}\mathbf{D}^j x_j\right)\mathbf{S}^{-1} + \sum_{j=1}^{D}\mathbf{W}_{\mathrm{n}}^j x_j. \qquad [S9]$$

Using $\sum_{j=1}^{D}\mathbf{D}^j x_j = \mathrm{diag}(\mathbf{p}_1^{\mathrm{T}}\mathbf{x},\cdots,\mathbf{p}_N^{\mathrm{T}}\mathbf{x})$ with $\mathbf{p}_i^{\mathrm{T}} = (\lambda_i^1,\cdots,\lambda_i^D)$, we rewrite Eq. **S8** as

$$\mathbf{u}(\mathbf{x}) = \mathbf{S}\begin{bmatrix} \exp(\mathbf{p}_1^{\mathrm{T}}\mathbf{x}) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \exp(\mathbf{p}_N^{\mathrm{T}}\mathbf{x}) \end{bmatrix}\mathbf{S}^{-1}\exp\left(\sum_{j=1}^{D}\mathbf{W}_{\mathrm{n}}^{j}x_j\right)\mathbf{u}(\mathbf{0}). \quad \text{[S10]}$$

This general solution is almost identical to the solution for diagonalizable weight matrices given in the main text (Eq. **6**) except for the exponential matrix $\exp(\sum_{j=1}^{D}\mathbf{W}_{\mathrm{n}}^{j}x_j)$, which is actually a polynomial in $\mathbf{x}$ because $\mathbf{W}_{\mathrm{n}}^{j}$ is a nilpotent matrix.

2.2. Fourier method. We start with the general solution in Eq. **6** of the main text,

$$\mathbf{u}(\mathbf{x}) = \sum_{i=1}^{N} c_i\mathbf{z}_i\exp\left(\mathbf{p}_i^{\mathrm{T}}\mathbf{x}\right), \quad \text{[S11]}$$

where $\mathbf{p}_i = (\lambda_i^1,\cdots,\lambda_i^D)^{\mathrm{T}}$ is a set of eigenvalues, with λ_i^k being the ith eigenvalue of matrix $\mathbf{W}^k$ and $\mathbf{z}_i$ the associated eigenvector. The coefficient $c_i = \mathbf{z}_i^{\mathrm{T}}\mathbf{u}(0)$ is completely determined and not a free parameter. For function approximation, there are a total of $N(N-1) + ND$ free parameters in Eq. **S11**: ND parameters for specifying the eigenvalues $(\mathbf{p}_1,\cdots,\mathbf{p}_N)$ and $N(N-1)$ parameters for specifying the normalized eigenvectors $(\mathbf{z}_1,\cdots,\mathbf{z}_N)$. The basis function set $\{\mathbf{z}_i\exp(\mathbf{p}_i^{\mathrm{T}}\mathbf{x})\}$ can be used to approximate any smooth function. For example, to obtain a sinusoid as a Fourier basis function, we can linearly combine a pair of exponential functions with conjugate imaginary eigenvalues. Because any smooth function can be at least approximated by a Fourier basis set, we limit ourselves to simultaneously diagonalizable matrices with imaginary eigenvalues, resulting in basis functions that are sinusoids. We first represent any given desired spatial patterns $\mathbf{u}(\mathbf{x}) = (u_1(\mathbf{x}),\cdots, u_N(\mathbf{x}))^{\mathrm{T}}$ for N units by a finite sum of sinusoids,

$$u_i(\mathbf{x}) = \sum_{j=1}^{n} A_{ij}\cos\left(\mathbf{q}_j^{\mathrm{T}}\mathbf{x}\right) + B_{ij}\sin\left(\mathbf{q}_j^{\mathrm{T}}\mathbf{x}\right), \quad \text{[S12]}$$

where $\mathbf{q}_1,\cdots,\mathbf{q}_n$ are distinct D-dimensional vectors that describe the orientations and spacings of the basis sinusoidal gratings and the scalar amplitudes A_{ij} and B_{ij} are readily obtained by the least-squares method. (Here $\mathbf{q}_i$ is related to the eigenvalues $\mathbf{p}_i$ in Eq. **S11** by $\mathbf{q}_i = \pm\sqrt{-1}\,\mathbf{p}_i$.) An equivalent vector formulation reads

$$u_i(\mathbf{x}) = \mathbf{c}_i^{\mathrm{T}}\mathbf{b}(\mathbf{x}), \quad \text{[S13]}$$

where $\mathbf{c}_i^{\mathrm{T}} = [A_{i1},\cdots,A_{in},B_{i1},\cdots,B_{in}]$ and $\mathbf{b}(\mathbf{x}) = [\cos(\mathbf{q}_1^{\mathrm{T}}\mathbf{x}),\cdots, \cos(\mathbf{q}_n^{\mathrm{T}}\mathbf{x}),\sin(\mathbf{q}_1^{\mathrm{T}}\mathbf{x}),\cdots,\sin(\mathbf{q}_n^{\mathrm{T}}\mathbf{x})]^{\mathrm{T}}$ is the basis set with $2n$ linearly independent sinusoids. To include all of the units, we write

$$\mathbf{u}(\mathbf{x}) = \begin{bmatrix} u_1(\mathbf{x}) \\ \vdots \\ u_N(\mathbf{x}) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{c}_1^{\mathrm{T}} \\ \vdots \\ \mathbf{c}_N^{\mathrm{T}} \end{bmatrix}}_{\mathbf{C}}\mathbf{b}(\mathbf{x}) = \mathbf{C}\mathbf{b}(\mathbf{x}). \quad \text{[S14]}$$

Next, to attain the network weights $\mathbf{W}^k$, we calculate the activity gradient as

$$\frac{\partial\mathbf{u}(\mathbf{x})}{\partial x_k} = \frac{\partial(\mathbf{C}\mathbf{b}(\mathbf{x}))}{\partial x_k} = \mathbf{C}\underbrace{\begin{bmatrix} \mathbf{0} & -\mathbf{Q}_k \\ \mathbf{Q}_k & \mathbf{0} \end{bmatrix}}_{\mathbf{S}_k}\mathbf{b}(\mathbf{x}) = \mathbf{C}\mathbf{S}_k\mathbf{b}(\mathbf{x}), \quad \text{[S15]}$$

where $\mathbf{Q}_k = \mathrm{diag}(q_{1k},\cdots,q_{nk})$ with $(q_{i1},\cdots,q_{iD})^{\mathrm{T}} = \mathbf{q}_i$, and we write the weighted activity as

$$\mathbf{W}^k\mathbf{u}(\mathbf{x}) = \mathbf{W}^k\mathbf{C}\mathbf{b}(\mathbf{x}). \quad \text{[S16]}$$

Because Eqs. **S15** and **S16** have to be equal ($\partial\mathbf{u}/\partial x_k = \mathbf{W}^k\mathbf{u}$) for all values of $\mathbf{b}(\mathbf{x})$, we find

$$\mathbf{C}\mathbf{S}_k = \mathbf{W}^k\mathbf{C}. \quad \text{[S17]}$$

To guarantee the existence of a solution, we assume that the number of units in the network is no less than the number of independent sinusoids ($n \geq 2n$) and that $\mathbf{C}$ has full rank (rank $\mathbf{C} = 2n$). Now a general solution to Eq. **S17** reads

$$\mathbf{W}^k = \mathbf{C}\mathbf{S}_k\mathbf{C}^{\dagger} + \mathbf{Y}(\mathbf{I}_N - \mathbf{C}\mathbf{C}^{\dagger}), \quad \text{[S18]}$$

where $\mathbf{C}^{\dagger}$ denotes the Moore–Penrose pseudoinverse, $\mathbf{Y}$ is an arbitrary $N \times N$ matrix, and $\mathbf{I}_N$ is the $N \times N$ identity matrix (2). The solution is not unique when $n > 2n$. Here we simply set $\mathbf{Y} = \mathbf{0}$ and choose the unique solution

$$\mathbf{W}^k = \mathbf{C}\mathbf{S}_k\mathbf{C}^{\dagger}. \quad \text{[S19]}$$

These weight matrices are guaranteed to commute because $\mathbf{C}^{\dagger}\mathbf{C} = \mathbf{I}_{2n}$ and $\mathbf{S}_k\mathbf{S}_l = \mathbf{S}_l\mathbf{S}_k$, so that

$$\mathbf{W}^k\mathbf{W}^l = \mathbf{C}\mathbf{S}_k\mathbf{C}^{\dagger}\mathbf{C}\mathbf{S}_l\mathbf{C}^{\dagger} = \mathbf{C}\mathbf{S}_k\mathbf{S}_l\mathbf{C}^{\dagger} = \mathbf{C}\mathbf{S}_l\mathbf{S}_k\mathbf{C}^{\dagger} = \mathbf{W}^l\mathbf{W}^k. \quad \text{[S20]}$$

In summary, given the desired activity patterns $\mathbf{u}(\mathbf{x})$, we approximate them in terms of a sinusoidal basis set $\mathbf{b}(\mathbf{x})$ to form the matrices $\mathbf{S}_1,\cdots,\mathbf{S}_D$ and to find the coefficient matrix $\mathbf{C}$. We then use Eq. **S19** to determine the network weights.

3. Examples of Commutativity of Weight Structure. In this section we consider several concrete examples that guarantee commutative weight matrices.

3.1. Commutativity of symmetric weights. Here we show that the symmetric weight scheme used in Fig. 3 of the main text guarantees commutativity of the weight matrices and hence path invariance. In this scheme, each unit is surrounded by 6 units spaced evenly on a circle of radius r, and this unit receives inputs from only the units that are on this surrounding circle (Fig. 3 and Fig. S2). The weight vector for the connection between 2 units is proportional to the vector connecting the spatial phases of the 2 units. As illustrated in Fig. S2*B*, the weight vector from unit j to unit i is

$$\mathbf{w}_{ij} = \begin{bmatrix} w_{ij}^1 \\ w_{ij}^2 \end{bmatrix} = K\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}, \quad \text{[S21]}$$

where the superscripts indicate spatial dimensions, K is a constant, and θ is the polar angle of the vector pointing from unit j to unit i with respect to the coordinate system (red axes in Fig. S2*B*). There is no self-connection ($w_{kk}^l = 0$). To verify the commutativity condition in Eq. **S7**, we rewrite it in component form,

$$\sum_{k=1}^{N} w_{ik}^1 w_{kj}^2 = \sum_{k=1}^{N} w_{ik}^2 w_{kj}^1, \quad \text{[S22]}$$

where the summation is over all $n = 9$ units in the network. By ignoring any difference in orientation, two-step indirect connections from any unit j to another unit i have only two possible patterns (Fig. S2*C*), depending on whether the shortest distance between the two units is r or $\sqrt{3}r$. In both cases there are two intermediate units a and b, so Eq. **S22** becomes

$$w_{ia}^1 w_{aj}^2 + w_{ib}^1 w_{bj}^2 = w_{ia}^2 w_{aj}^1 + w_{ib}^2 w_{bj}^1. \quad \text{[S23]}$$

Because $\mathbf{w}_{ia} = \mathbf{w}_{bj}$ and $\mathbf{w}_{ib} = \mathbf{w}_{aj}$ by the way the weight vectors are specified, we can rewrite the left-hand side of Eq. **S23** as $w_{bj}^1 w_{ib}^2 + w_{aj}^1 w_{ia}^2$, which is indeed equal to the right-hand side.

3.2. Other special examples. Here we discuss two special network structures that guarantee path invariance. For the first example,

we suppose the network dynamics in Eq. **1** (or Eq. **S3**) are state independent, namely, $\mathbf{F}(\mathbf{u}) = \mathbf{M}$, where $\mathbf{M} = \{m_{ik}\}$ ($i = 1,\cdots, N$ and $k = 1,\cdots, D$) is an arbitrary constant matrix. This system satisfies the equality of mixed partial derivatives from Eq. **2** (or Eq. **S6**) because they all vanish. The activity of each unit i depends linearly on location $\mathbf{x}$ as given by the dot product $u_i = \sum_{k=1}^{D} m_{ik}x_k = \mathbf{m}_i^{\mathrm{T}}\mathbf{x}$ with $\mathbf{m}_i = (m_{i1},\cdots, m_{iD})^{\mathrm{T}}$.

The second example is a linear system (Eq. **3**) with all identical weights $\mathbf{W}^1 = \mathbf{W}^2 = \cdots \mathbf{W}^D$, which trivially commute and satisfy Eq. **5** (or Eq. **S7**). We can drop the subscripts in the eigenvalues and write the eigenvalue vector in Eq. **S11** (or Eq. **6**) as $\mathbf{p}_i = \lambda_i\mathbf{1}$, where $\mathbf{1} = (1,\cdots, 1)^{\mathrm{T}}$ has D entries. The solution in Eq. **S11** now becomes $\mathbf{u}(\mathbf{x}) = \sum_{i=1}^{N} c_i\mathbf{z}_i \exp(\lambda_i\mathbf{1}^{\mathrm{T}}\mathbf{x})$. Note that the activity stays unchanged: $\mathbf{u}(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{u}(\mathbf{x})$, for any displacement $\Delta\mathbf{x}$ perpendicular to the vector $\mathbf{1}$ because $\mathbf{1}^{\mathrm{T}}(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{1}^{\mathrm{T}}\mathbf{x}$. Thus, the dynamics collapse onto a single dimension given by $\mathbf{1}$, and the spatial activity pattern $\mathbf{u}$ varies only along this axis.

4. Linear System Learning Rule. The desired weights of the linear network in Eq. **3** of the main text can be learned by a local supervised learning rule starting from random initial weights. During learning, we assume that the desired network activity pattern $\mathbf{s}$ as a function of location $\mathbf{x}$ is set up by other cues such as landmarks without relying on the path integration mechanism. Define an error function as $E = \frac{1}{2}(\dot{\mathbf{s}} - \dot{\mathbf{u}})^{\mathrm{T}}(\dot{\mathbf{s}} - \dot{\mathbf{u}})$, where $\dot{\mathbf{s}}$ is the desired time derivative and $\dot{\mathbf{u}} = (\sum_{j=1}^{D}\mathbf{W}^j v_j)\mathbf{u}$ is the actual time derivative given by Eq. **3** of the main text. Gradient descent on the error function yields the learning rule

$$\Delta w_{ij}^k = -\alpha\frac{\partial E}{\partial w_{ij}^k} = \alpha v_k u_j(\dot{s}_i - \dot{u}_i), \qquad \textbf{[S24]}$$

where $\alpha > 0$ is the learning rate. Depending on the instantaneous velocity, the weight matrices that line up the most with that direction will be updated the most at that time. As long as all directions are sufficiently sampled, proper weights will be learned. The solution is not unique if the network is too large, as expected from Eq. **S18**. An example showing convergence of the weights to the proper values is shown in Fig. S1.

5. Examples of Exact Integrators. In the following we examine four existing examples of exact path integrators and confirm that they are indeed equivalent to the form given by Eq. **S3** with condition Eq. **S6** also satisfied.

Example 1: Linear eye position integrator. For a linear eye position integrator model, its dynamics obeys the differential equation

$$\tau\frac{du_i}{dt} = -u_i + \sum_{j=1}^{N} w_{ij}u_j + c_i v, \qquad \textbf{[S25]}$$

where u_i is the activity of the ith neuron, w_{ij} is the connection weight from neuron j to neuron i, and $v = dE/dt$ is the eye velocity signal from the vestibular afferents with E being the eye position (3, 4). The weight matrix w_{ij} is supposed to have an eigenvalue $\lambda_1 = 1$ whereas the remaining eigenvalues all have real parts <1. Once the system settles into the eigenstate associated with the eigenvalue $\lambda_1 = 1$, we can reduce Eq. **S25** to the form

$$\frac{du}{dt} = kv, \qquad \textbf{[S26]}$$

where $\mathbf{u}$ is the amplitude for that eigenstate and k is a constant coefficient. This equation means that $\mathbf{u}$ is a linear function of the eye position E. Eq. **S26** is a special case of Eq. **S3** in one-dimensional space. Although in general Eq. **S25** is not an exact integrator, it becomes one after it has settled into its stable eigenstate, and the system becomes consistent with our general equation.

Example 2: Continuous attractor recurrent network. Attractor networks have been used to model head direction cells, place cells, and grid cells (5, 6). Here we consider the activity pattern $u = u(\mathbf{x}, t)$ of a continuum of neurons as a function of time t and a continuous spatial index $\mathbf{x}$ that specifies the neurons in the continuum (7). Assuming that the connection weights are shift invariant, namely, the connection between two neurons depends only on their relative locations, we have the dynamical equation

$$\tau\frac{\partial u}{\partial t} = -u + W * g(u), \qquad \textbf{[S27]}$$

where τ is a time constant, g is the gain function that describes the input–output relation of individual neurons, and $*$ indicates convolution with respect to the spatial index. When the weight pattern is symmetric, the system allows a stationary activity bump that satisfies the equation

$$-u + W * g(u) = 0. \qquad \textbf{[S28]}$$

The modified equation below is an exact integrator that can keep the same shape of the activity bump as in the stationary case while the bump moves at an instantaneous velocity $\mathbf{v}$,

$$\tau\frac{\partial u}{\partial t} = -u + (W - \tau\mathbf{v}\cdot\nabla W) * g(u), \qquad \textbf{[S29]}$$

where $\nabla = (\partial/\partial x_1,\cdots, \partial/\partial x_D)$ is the gradient (8). At first glance, this equation looks very different from our general Eq. **S3**. To see their relationship, note that because the traveling activity bump has identical shape to the stationary activity bump, it also satisfies Eq. **S28**. Subtracting this equation from Eq. **S29**, we obtain

$$\frac{\partial u}{\partial t} = -\mathbf{v}\cdot\nabla(W * g(u)), \qquad \textbf{[S30]}$$

where we have used the following identity for convolution: $(\nabla W) * g(u) = \nabla(W * g(u))$. Using Eq. **S28** again, we can rewrite Eq. **S30** as

$$\frac{\partial u}{\partial t} = -\mathbf{v}\cdot\nabla u. \qquad \textbf{[S31]}$$

Except for the negative sign, this equation is equivalent to the derivative chain rule in Eq. **S1** from which our general theory is derived.

The negative sign in Eq. **S31** is due to the fact that here the derivative $\nabla = \partial/\partial\mathbf{x}$ is with respect to the index $\mathbf{x}$ of the neuronal population, whereas the derivative in our general theory is with respect to the encoded variable, which is denoted by $\hat{\mathbf{x}}$ here to make a distinction. Let $\phi(\mathbf{x})$ describe the stereotyped shape of the activity bump center at $\mathbf{0}$. Then the activity bump centered at the true value of the encoded variable $\hat{\mathbf{x}}$ is $u = \phi(\mathbf{x} - \hat{\mathbf{x}})$. It follows that $\partial u/\partial\mathbf{x} = -\partial u/\partial\hat{\mathbf{x}}$, which accounts for the flip of sign.

The continuous attractor network in Eq. **S29** differs from the standard form of our general theory, and the system cannot perform exact path integration before its transient activity has reached a stable state. Once the system has settled into a stable traveling activity bump with stereotyped shape, the dynamics of the continuous attractor network become an exact integrator and have exactly the same form as required by our general theory, as shown above.

Example 3: Interference of velocity-modulated oscillators. In oscillatory interference models (9–11), the total synaptic input to a neuron (such as a grid cell or a place cell) is a sum of the activities of n theta oscillators,

$$I = \sum_{k=1}^{n} w_k \cos\phi_k, \qquad \text{[S32]}$$

where ϕ_k is the phase of oscillator k and w_k is the weight. The instantaneous frequency of each oscillator k is given by $d\phi_k/dt$ and is assumed to have the form

$$\frac{d\phi_k}{dt} = \omega + \mathbf{p}_k \cdot \mathbf{v}, \qquad \text{[S33]}$$

where ω is the base theta frequency that is common among all oscillators although it may vary with time, vector $\mathbf{v}$ is the instantaneous running velocity, and each oscillator has a preferred running direction that is specified by the direction of the vector $\mathbf{p}_k$. There are already some experimental results in support of Eq. **S33** (11).

This system is an exact path integrator in the sense that the envelope of the signal given by Eq. **S32** has the same spatial profile regardless of the trajectory taken. By the envelope rule (12), the envelope can be written as

$$E = \left| \sum_{k=1}^{n} w_k \exp(i\phi_k) \right| = \sqrt{\sum_{k=1}^{n}\sum_{l=1}^{n} w_k w_l \cos(\phi_k - \phi_l)}, \qquad \text{[S34]}$$

where the common carrier wave described by the common frequency ω has disappeared. By integrating Eq. **S33** from time 0 to time t, we obtain

$$\phi_k(t) - \phi_l(t) = \phi_k(0) - \phi_l(0) + (\mathbf{p}_k - \mathbf{p}_l) \cdot [\mathbf{x}(t) - \mathbf{x}(0)], \qquad \text{[S35]}$$

where $\mathbf{x}(t)$ and $\mathbf{x}(0)$ are the animal's location at time t and time 0, respectively. Thus, the final envelope in Eq. **S34** depends on the final position $\mathbf{x}(t)$ but not the actual trajectory.

This system is consistent with our general theory because for each given $u \equiv \phi_k - \phi_l$ we have

$$\frac{du}{dt} = \mathbf{p} \cdot \mathbf{v}, \qquad \text{[S36]}$$

where $\mathbf{p} \equiv \mathbf{p}_k - \mathbf{p}_l$ is a constant vector. This is a trivial case of our general Eq. **S3**, and condition Eq. **S6** is also satisfied because the derivative of a constant vector always vanishes.

Example 4: Calculating head-centered gravity from angular velocity. Representation of head orientation in 3D space may be performed by integrating head angular velocity $\boldsymbol{\omega}$ (from vestibular afferents) to calculate head-centered gravity $\mathbf{g}$ using the following equation (13):

$$\frac{d\mathbf{g}}{dt} = -\boldsymbol{\omega} \times \mathbf{g}. \qquad \text{[S37]}$$

This equation does not appear to conform to our invariance conditions because the mixed partial derivatives are unequal, and yet it still performs accurate integration. This problem arises because angular velocity $\boldsymbol{\omega}$ is an inexact differential quantity and rotations do not commute. For example, a roll followed by a pitch is different from a pitch followed by a roll. Nonetheless, this model comes into agreement with our theory if we convert it to a path-invariant coordinate system by using Euler angles (ϕ, θ, ψ) that describe head orientation relative to a fixed reference frame. Plugging

$$\boldsymbol{\omega} = \begin{bmatrix} \sin\psi\sin\theta & \cos\psi & 0 \\ \cos\psi\sin\theta & -\sin\psi & 0 \\ \cos\theta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \qquad \text{[S38]}$$

into Eq. **S37** yields

$$\frac{d\mathbf{g}}{dt} = \begin{bmatrix} -g_z\cos\psi\sin\theta + g_y\cos\theta & g_z\sin\psi & g_y \\ g_z\sin\psi\sin\theta + g_x\cos\theta & g_z\cos\psi & -g_x \\ -g_y\sin\psi\sin\theta + g_x\cos\psi\sin\theta & -g_y\cos\psi - g_x\sin\psi & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}, \qquad \text{[S39]}$$

with $(g_x, g_y, g_z)^{\mathrm{T}} = \mathbf{g}$, which can be further expressed as

$$\mathbf{g} = \begin{bmatrix} \cos\psi\cos\phi - \cos\theta\sin\phi\sin\psi & \cos\psi\sin\phi + \cos\theta\cos\phi\sin\psi & \sin\psi\sin\theta \\ -\sin\psi\cos\phi - \cos\theta\sin\phi\cos\psi & -\sin\psi\sin\phi + \cos\theta\cos\phi\cos\psi & \cos\psi\sin\theta \\ \sin\theta\sin\phi & -\sin\theta\cos\phi & \cos\theta \end{bmatrix} \mathbf{g}_0, \qquad \text{[S40]}$$

where $\mathbf{g}_0$ is the value of $\mathbf{g}$ with respect to the reference frame when $(\phi, \theta, \psi) = (0, 0, 0)$. By setting $\mathbf{u} = \mathbf{g}$ and $\mathbf{x} = (\phi, \theta, \psi)^{\mathrm{T}}$, now Eq. **S39** conforms to Eq. **S3** and, with a quick check of the mixed partial derivatives, fulfills Eq. **S6** as well. Although Euler angles are sufficient for our purpose here, they are by no means a unique coordinate system to show compatibility with the general theory; other coordinate systems may work as well. The consideration above demonstrates an important caveat: Although an exact path integrator must follow the conditions of multiplicative modulation and equality of mixed partial derivatives, these computations are performed with respect to a proper coordinate system but not necessarily for arbitrary variables (e.g., angular velocity).

6. Remarks on Transient Decay in a Path Integrator. One may wonder why the basic dynamical equation in the main text (Eq. **1**) does not contain a decay term $-\mathbf{u}$ as commonly seen in neural models. There are at least two possible ways for accommodating a decay term in the dynamics while still maintaining consistency with Eq. **1**. We demonstrate the first possibility by considering the linear case of Eq. **1** as given by Eq. **3** of the main text,

$$\frac{d\mathbf{u}}{dt} = \mathbf{W}^{\mathrm{eff}}\mathbf{u}, \qquad \text{[S41]}$$

where $\mathbf{W}^{\mathrm{eff}} = \sum_{k=1}^{D} \mathbf{W}^k v_k$ is the effective weight matrix, with $\mathbf{v} = (v_1, \cdots, v_D)^{\mathrm{T}}$ being the velocity. Because Eq. **S41** is equivalent to

$$\frac{d\mathbf{u}}{dt} = -\mathbf{u} + \left(\mathbf{W}^{\mathrm{eff}} + \mathbf{I}\right)\mathbf{u}, \qquad \text{[S42]}$$

where $\mathbf{I}$ is the $N \times N$ identity matrix, we have incorporated the decay term $-\mathbf{u}$ here without altering the actual dynamics. Here adding a constant (velocity-independent) diagonal matrix to the original effective weight matrix $\mathbf{W}^{\mathrm{eff}}$ is equivalent to adding excitatory self-connections (autapse) to all units in the network.

The second possibility is more intriguing. Note that in *SI Text 5*, *Example 1: Linear eye position integrator*, the dynamical equation (Eq. **S25**) contains a decay term ($-u_i$), but after the system settles into an equilibrium state, the new equation (Eq. **S26**) no longer has any decay term. Similarly, in *SI Text 5*, *Example 2: Continuous attractor recurrent network*, the original nonlinear dynamical equation (Eq. **S27**) also contains a decay term ($-u$)

but, when an equilibrium state is reached, the decay term vanishes from Eq. **S30** (or Eq. **S31**). In both examples above, the original dynamical systems cannot perform path integration exactly unless the transient dynamics due to the decay terms have already decayed. Only when the systems settle into their equilibrium states (attractor states) do they begin to behave as exact path integrators that satisfy Eq. **1** of the main text. Therefore, in general Eq. **1** (or Eq. **3** for the linear case) may represent the desired dynamics of an exact integrator only after they have already reached an equilibrium state, whereas the original dynamics (allowing transient decays) can be more complicated. In the main text we have considered only the most straightforward interpretation of Eq. **3** (or Eq. **S41**) without using any decay term or more complicated dynamics.

7. Simulation Parameters. Differential equations were simulated in MATLAB (MathWorks) using the ode113 function, which is a variable-order Adams–Bashforth–Moulton Prediction-Evaluation-Correction-Evaluation (PECE) solver. The time step for the grid cell simulation was set at 18 ms for display purposes. Default error tolerances were used. Animal trajectory was simulated by generating a uniform random variable, convolving with a 25-point rectangular window, and then scaling so that the maximum velocity is <50 cm/s.

1. Edwards HM (1994) *Advanced Calculus: A Differential Forms Approach* (Birkhäuser, Boston).
2. Horn RA, Johnson CR (1990) *Matrix Analysis* (Cambridge Univ Press, Cambridge, UK).
3. Seung HS (1996) How the brain keeps the eyes still. *Proc Natl Acad Sci USA* 93: 13339–13344.
4. Cannon SC, Robinson DA, Shamma S (1983) A proposed neural network for the integrator of the oculomotor system. *Biol Cybern* 49:127–136.
5. McNaughton BL, Battaglia FP, Jensen O, Moser EI, Moser M-B (2006) Path integration and the neural basis of the 'cognitive map'. *Nat Rev Neurosci* 7:663–678.
6. Knierim JJ, Zhang K (2012) Attractor dynamics of spatially correlated neural activity in the limbic system. *Annu Rev Neurosci*, 10.1146/annurev-neuro-062111-150351.
7. Wilson HR, Cowan JD (1972) Excitatory and inhibitory interactions in localized populations of model neurons. *Biophys J* 12:1–24.
8. Zhang K (1996) Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: A theory. *J Neurosci* 16:2112–2126.
9. Burgess N, Barry C, O'Keefe J (2007) An oscillatory interference model of grid cell firing. *Hippocampus* 17:801–812.
10. Monaco JD, Knierim JJ, Zhang K (2011) Sensory feedback, error correction, and remapping in a multiple oscillator model of place-cell activity. *Front Comput Neurosci*, 10.3389/fncom.2011.00039.
11. Welday AC, Shlifer IG, Bloom ML, Zhang K, Blair HT (2011) Cosine directional tuning of theta cell burst frequencies: Evidence for spatial coding by oscillatory interference. *J Neurosci* 31:16157–16176.
12. Hartmann WM (1997) *Signals, Sound, and Sensation* (Springer, New York).
13. Green AM, Angelaki DE (2007) Coordinate transformations and sensory integration in the detection of spatial orientation and self-motion: From models to experiments. *Prog Brain Res* 165:155–180.

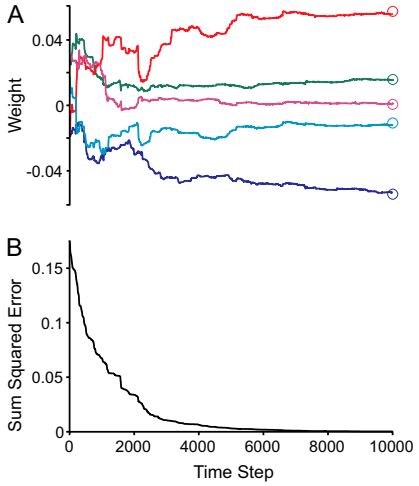

Fig. S1. Convergence of weights in a linear network by gradient descent learning. We learned the weights used in the network in Fig. 2 by the learning rule in Eq. **S24**. (*A*) Five representative weights are shown out of a total of 72 component weights. Each target weight is denoted by a circle at the endpoint. Weights were initialized at random values. (*B*) The sum of the squared error for all weights is shown as a function of time. The error gradually approached zero, indicating the convergence of all of the weights to their proper values.

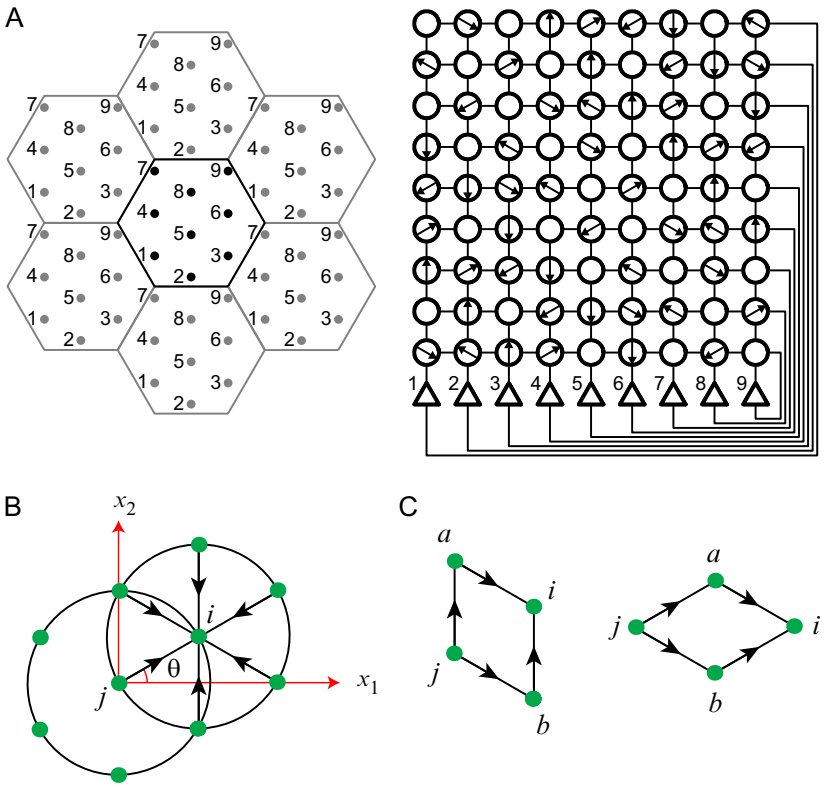

Fig. S2. Grid cell network with radially symmetric connections always have commutative weight matrices. (*A*) The spatial phases (left) and full weight connection structure (right) of the network from Fig. 3 of the main text are shown. Note the regularity in weight patterns. (*B*) The weight structure is radially symmetric. Each unit receives inputs only from a surrounding ring of units, with the weight vector proportional to the displacement between the two units (arrows). (*C*) The pattern of intermediate connections between any two units *i* and *j* falls into one of the two types, up to a rotation.

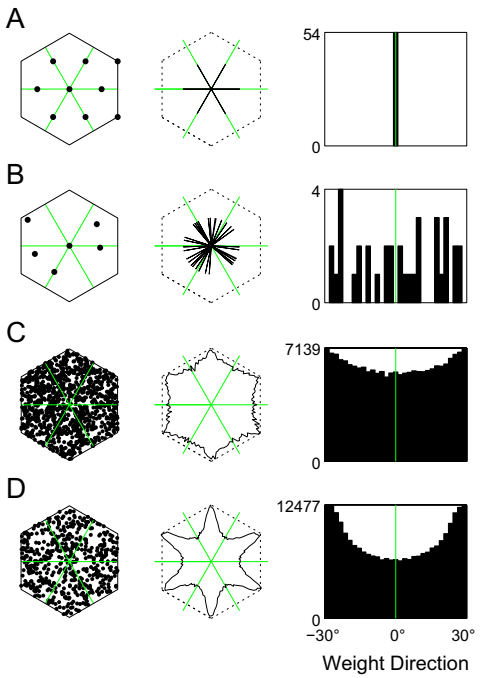

Fig. S3. Distributions of weight vector directions in various grid cell networks. The left column shows the spatial phases within a single grid tile for all of the units in each network. The middle column shows either individual weight vector directions (top two cases) or polar distributions of large numbers of weight vector directions (bottom two cases). The right column shows this distribution across 60 degrees of rotational symmetry with respect to the green lines, which are aligned with the grid. (*A*) For evenly spaced spatial phases (same 9-cell network as Figs. 3 and S1 but rotated by 30 degrees), all 54 nonzero weight vectors line up to the grid. (*B*) For a single example of a 6-cell network with random spatial phases (same network as Fig. 2), the directions of 36 weight vectors obtained from Eq. **S19** are distributed roughly evenly. (*C*) For a large number (5000) of 6-cell networks with random spatial phases chosen from a uniform distribution on a hexagonal tile, a pattern emerges where there is a weak preference for directions out of phase with the grid. (*D*) For a single network with 500 cells, the weights solved from Eq. **S19** show an even stronger preference for directions out of phase with the grid, in stark contrast to the network in *A*.