What Grid Cells Convey about Rat Location

Ila R. Fiete,[1,3] **Yoram Burak,**[1,4] **and Ted Brookings**[2,5]

[1]Kavli Institute for Theoretical Physics and [2]Department of Physics, University of California, Santa Barbara, Santa Barbara, California 93106, [3]Broad Fellows Program, Computation and Neural Systems, California Institute of Technology, Pasadena, California 91125, [4]Swartz Fellows Program, Center for Brain Science, Harvard University, Cambridge, Massachusetts 02138, and [5]Volen Center, Department of Biology, Brandeis University, Waltham, Massachusetts 02454

We characterize the relationship between the simultaneously recorded quantities of rodent grid cell firing and the position of the rat. The formalization reveals various properties of grid cell activity when considered as a neural code for representing and updating estimates of the rat's location. We show that, although the spatially periodic response of grid cells appears wasteful, the code is fully combinatorial in capacity. The resulting range for unambiguous position representation is vastly greater than the $\approx$1–10 m periods of individual lattices, allowing for unique high-resolution position specification over the behavioral foraging ranges of rats, with excess capacity that could be used for error correction. Next, we show that the merits of the grid cell code for position representation extend well beyond capacity and include arithmetic properties that facilitate position updating. We conclude by considering the numerous implications, for downstream readouts and experimental tests, of the properties of the grid cell code.

Key words: navigation; dMEC; entorhinal; hippocampus; spatial perception; information; path integration; theory

Introduction

The brain uses diverse schemes to encode information. For example, oculomotor neurons and head direction neurons both encode continuous one-dimensional (1-D) variables but in very different ways: oculomotor neurons encode the horizontal angle of gaze in vertebrates through a rate-proportional code, all firing together at rates that increase linearly with temporal deflections of the eye (Lopez-Barneo et al., 1982). Neurons in the rodent subiculum encode where the animal is facing by firing when its head points in a narrow angular range, with different neurons tuned to different head directions (Taube et al., 1990). In insects, antennal lobe neurons encode the identity of odors with distributed, temporally evolving patterns of firing, whereas Kenyon cells of the mushroom body encode similar information using a more sparse, static response (Perez-Orive et al., 2002). Hippocampal place cells encode a rat's whereabouts by firing only when the rat is within a neighborhood of a particular location, with the preferred location varying from cell to cell (O'Keefe and Nadel, 1978).

In contrast with place cells, so-called grid cells, neurons in the dorsolateral band of the medial entorhinal cortex (dMEC), fire whenever the rat, moving freely in a two-dimensional (2-D) enclosure, is on any vertex of a virtual triangular lattice overlaid on the surface of the enclosure (Fyhn et al., 2004; Hafting et al., 2005). The position-dependent firing of these cells, as well as the fact that entorhinal cortex is the primary cortical input to the hippocampus (which plays an important role in spatial memory and navigation) suggest that grid cells may be involved in encoding an internal estimate of rat position (Mittelstaedt and Mittelstaedt, 1980; Whishaw and Maaswinkel, 1998; Etienne and Jeffery, 2004; Hafting et al., 2005; Burak and Fiete, 2006; Fuhs and Touretzky, 2006).

However, the spatially periodic nature of grid cell firing is puzzling: the activity of a cell does little to specify the rat's location, because it fires when the rat is near any of the vertices of the virtual lattice. Furthermore, the periodicity of any of the grids is not large enough to provide unambiguous information about rat position: a rat may forage over distances of 100 m to 1 km in the wild (Recht, 1988; Miller and Clesceri, 2002; Russell et al., 2005), whereas the measured lattice periods are no more than 1–10 m. It appears profligate at best or impossible at worst, to represent a large-ranging, nonperiodic variable such as rat position by a set of periodically repeating variables, each with very small periods. Furthermore, if dMEC is the idiothetic path integrator, as has been hypothesized (Hafting et al., 2005; Burak and Fiete, 2006; Fuhs and Touretzky, 2006; McNaughton et al., 2006), it must represent position with high resolution to minimize the accumulation of rounding off errors. These observations provoke questions about exactly how much positional information is unambiguously stored within the grid cells, why it is coded the way it is, and the ways in which such information may be used by downstream areas.

Here we consider the representational properties of the grid cell response, considered as a code for position. Our purpose is not to provide a theory or model of how grid cell activity patterns

Table 1. Capacity and "arithmeticity" of the modulo code: comparison with other systems

Representation	Capacity	Algorithmic rules for: rep(x)	x+y	x>y?	Carry-free	Narrow range
Unary	~N	✓	✓	✓	—	—
Roman numeral	~e^N	✓	✗	✓	—	—
Combinatorial name	~e^N	✓	✗	✗	—	—
Fixed-base	~e^N	✓	✓	✓	✗	✗
Modulo	~e^N	✓	✓	✗	✓	✓

"Capacity" refers to how capacity scales. "Algorithmic rules" refers to the ability to represent and compute the corresponding quantities or inequalities (x, x + y, and x > y) with simple rules and without a lookup table. "Carry-free" refers to whether arithmetic operations may be performed without carrying over information from one register to the next. "Narrow range" refers to whether different registers represent similar size scales. The last two criteria only apply to representational schemes with register structures and algorithmic rules for representing arithmetic operations. ✓, Yes; X, no; —, not applicable .

are generated as the rat moves around (for such studies, see Burak and Fiete, 2006; Fuhs and Touretzky, 2006; McNaughton et al., 2006). Rather, we take as given the simultaneous experimental measurements of rat position and neural activity and characterize the relationship between the two recorded quantities. This characterization allows us to understand the properties of capacity, arithmeticity, and robustness of position representation in grid cells. It also suggests possibilities and limits for the readout of entorhinal cortical codes by the hippocampus.

A shorter, preprint version of this article has been published previously (Burak et al., 2006).

Materials and Methods

In this section, we define some central concepts, establish terminology, and outline the experimental findings on which the results of this work are based.

Representation schemes

On what basis does the brain select one specific encoding scheme from among various candidate possibilities? A simple analogy, from the numeral representation of integers, demonstrates that different schemes can have intrinsically different storage and computational properties.

Capacity. Integers may be written in any base-*b* positional numeral system (e.g., *b* = 2 for binary, *b* = 10 for decimal, etc.) or more generally may be represented in systems that do not use a fixed base at all. One such scheme is to represent the number 8 by eight ticks. This scheme, known as a unary code, requires an alphabet of exactly one coding symbol or letter (a tick), but the number of repetitions of that element ("word length") needed to represent a number grows in direct proportion to its size: representational capacity grows only linearly with word length. By comparison, base-*b* positional numeral systems have a major advantage over this scheme: an alphabet of *b* letters can be combined into words of length N to represent integers from 0 to $b^N - 1$, a range that increases exponentially as a function of the word length N (Table 1). These examples demonstrate that representational capacity can vary dramatically with the choice of encoding scheme.

"Arithmeticity." The choice of encoding scheme can also determine whether it is possible (and, if so, how easy it is) to extract metric information from a number or to perform arithmetic operations on it. In the unary encoding scheme, it is easy to compare the sizes of two numbers directly from their representations. These tasks are also relatively easy in the positional fixed-base numeral system. In contrast, consider a scheme in which each number is represented by a unique, distinct letter (similar to a neural "grandmother cell" code), or a combinatorial "label" scheme in which each number is represented by a name or label, a combination of letters with no metric structure. In either case, it is impossible to judge the relative sizes of the represented numbers without an explicit lookup table (Table 1).

The ability to simply and algorithmically perform basic arithmetic operations such as addition or multiplication of numbers within the representational scheme is another arithmetic property of an encoding scheme. The roman numeral system approaches the compactness of a fully combinatorial code and allows for relatively easy magnitude comparison; however, it is not possible to write simple rules for computing the sum of two numbers within this system. The situation is worse for combinatorial labels. In contrast, fixed-base numeral systems enable arithmetic operations such as addition and multiplication of pairs of large numbers using simple rules (increment, carry) to produce a result in the same base (Table 1).

In these examples, the choice of an encoding scheme had profound implications for representational capacity and arithmeticity. Similarly, the specific choice made for the representation of a variable in the brain is likely to be informative about the constraints and functional priorities underlying the local neural computation and its downstream uses.

From the point of view of position representation, capacity is important if the range of positions to be represented is large and if the required resolution is high. Arithmeticity is important if the representation is used to perform incremental position updating as the rat moves around, an intrinsically arithmetic operation on the metric quantity of position.

Salient properties of grid cells

In recent experiments (Fyhn et al., 2004; Hafting et al., 2005), rats foraged in 2-D enclosures searching for randomly scattered food pellets while rat position and the activity of single neurons in dMEC was monitored.

Single cell. Single-neuron activity, when plotted as a function of rat position, forms a triangular lattice. An individual neuron fires whenever the rat is in the neighborhood of any vertex of an imagined regular triangular lattice, overlaid on the explored space (Fig. 1*A*, left). The radius of each of the periodically arranged firing fields is quite large, extending over one-fifth to one-third of the lattice period.

Because the rat repeatedly visits the same location using different routes, which elicits a similar position-based grid cell response, the activity of grid cells is independent of the rat's path.

The periodicity of the grid is reported to be independent of the shape or size of the enclosure in which the rat is running (Fig. 1*A*, right) (Hafting et al., 2005) [although temporary rescaling may be observed if a familiar enclosure is suddenly resized (Barry et al., 2007)]. The periodic response tiles the entire explored area without noticeable distortion of the lattice close to the boundaries of the enclosure. Based on these observations, we assume that a similar response will also be observed in far larger enclosures than used so far in experimental assays.

Cell population, single lattice. Neighboring neurons in dMEC all respond to rat position with the same lattice period (λ) and orientation but typically differ in their relative spatial phases (Hafting et al., 2005) (Fig. 1*B*). Together, these neurons cover all possible lattice phases. In other words, at any rat position, some subset of these neurons is active, the identity of which can specify the phase of the rat's position within a unit cell of the lattice.

As a population, all the neurons with the same period cannot specify anything about which unit cell the rat is in. The observed population response is identical for all positions separated by any integer multiple of the lattice period along either principal lattice direction (Fig. 1*D*).

Mathematically, the information represented by the entire local neural population may be described simply as a 2-D phase within a unit cell of the triangular lattice. Later, we assume that this phase can only be ascertained up to a certain accuracy, related to the size of the firing field at each vertex.

Cell population, multiple lattices. Neurons separated by larger distances along dMEC have different lattice periods (Fig. 1*C*). The smallest measured period (at the dorsal end) is ≈30 cm, whereas the largest measured period (halfway to the ventral boundary) is ≈70 cm (Hafting et al., 2005). By extrapolating (linearly or exponentially) the monotonic trend in lattice periods up to the ventral boundary, the largest lattice period may be 1–10 m.

The radius of the firing field at each lattice vertex scales in direct proportion to the lattice period, so that the larger period lattice responses look like globally stretched versions of the responses of cells with smaller periods.
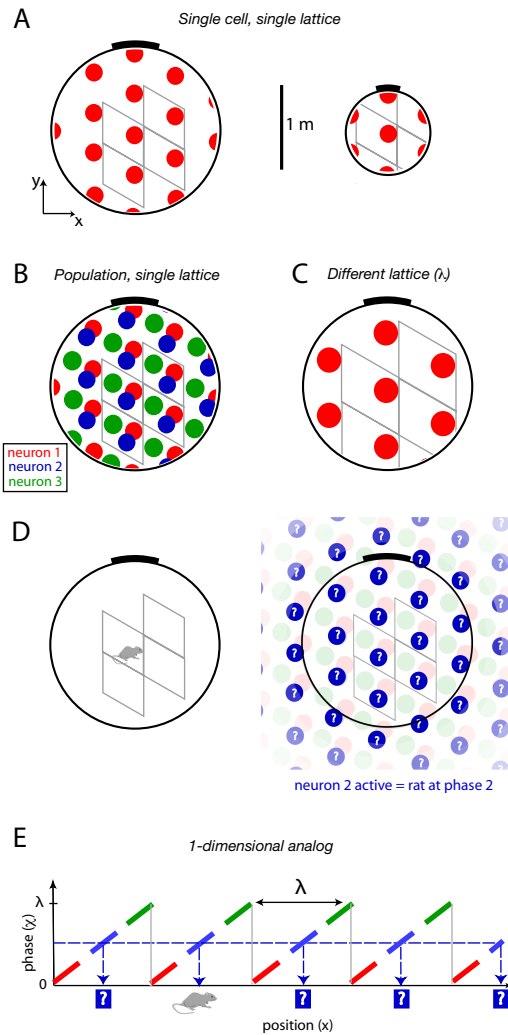
Figure 1. Schematic of mathematically salient grid cell properties. **A**, Single-cell activity. Left, The firing response of one dMEC neuron as a function of real-space rat position, obtained by summing the response of the cell over time while the rat explores a circular enclosure. Red, Cell firing. White, Cell quiescent. The cell fires when the rat visits the vicinity of any vertex of a virtual regular triangular lattice tiling space. The lattice is characterized by its period (blob spacing), angular orientation (lattice rotation), and phase (lattice translations, which are unique up to one unit cell). Gray diamonds, Unit cells of the lattice. Right, The lattice period is reported to be independent of the size or shape of the enclosure. **B**, The responses of neighboring cells (red, blue, and green) share a single lattice period and angular orientation. Their firing only differs by lattice translations or phase. **C**, At larger separations in dMEC, the responses of different neurons have different spatial periods. **D**, The instantaneous rat position (left) corresponds to the firing of one set of neurons (blue) in a dMEC lattice. The firing of these cells specifies rat position as a phase within any unit cell of the population lattice (right), but not which unit cell the rat is in. **E**, One-dimensional analog: the rat may be at location x_0 or at any location $x_0 + m\lambda$, separated by integer multiples (m) of λ (lattice period) from that location, consistent with the phase corresponding to the firing of the "blue" cell.

Results

To summarize all the observations in Materials and Methods, dMEC decomposes the 2-D vector of instantaneous, time-varying rat position into a set of time-varying 2-D periodic variables, or phases. Each phase is ambiguous about rat position up to all translations in the infinite plane by any integer multiple of the period of that lattice, λ, along the principal lattice directions.

Formal mathematical description in one dimension
For simplicity, we consider below a purely 1-D analog of the dMEC representation, described above. Our conclusions on capacity

arithmeticity carry over directly to the 2-D case, and we believe our conclusions on capacity will, too (see Discussion). In the 1-D analog, the identity of the active subpopulation within each lattice population α represents a 1-D phase χ_α given by the following:

$$\chi_\alpha(x) = x \bmod \lambda_\alpha. \tag{1}$$

where x is the rat's internal estimate of its position in space. When the space is filled with sensory cues, x is essentially the rat's true location; if the rat has been moving about in a landmark-free space, x may be a relatively poor estimate of its location. As shorthand in what follows, we will frequently refer to x as the rat's position, although by x we always mean the rat's internal estimate of its position; similarly, we will refer to the rat's estimate of its position by the shorthand of "the rat's position." λ_α is the lattice period of lattice α. The symbol mod represents the modulo operation, which is the remainder after integer division of x by λ_α (Fig. 1 E), and, without loss of generality, the phases have been defined to be 0 at $x = 0$.

Because there are N different neural populations with different lattice periods ($\lambda_1, \lambda_2, \ldots \lambda_N$), each position x is represented in dMEC by the set of phases $\vec{\chi} = (x \bmod \lambda_1, x \bmod \lambda_2, \ldots x \bmod \lambda_N)$, which can be read out as the identity of the active subpopulation within each lattice (Fig. 1 D,E).

Interestingly, the representation of a quantity by a set of remainders (phases) after modulo division with a fixed set of moduli (lattice periods), as is happening in dMEC, constitutes an unusual yet well known numeral system for the representation of numbers, called the residue number system (RNS) (Soderstrand et al., 1986). Although it may seem like a strange concept that the brain might implement a formal numeral system to represent position, position is after all a metric quantity and one that must be updated arithmetically as the rat moves. We will see that the modulo code for position shares some of the best features of the base-b numeral systems for the representation of numeric quantities (Table 1) and has additional properties that are more advantageous in the neurobiological context.

Capacity
The one-to-many mapping from phase to position (Fig. 1 D,E), described in Materials and Methods and formalized above, means that the phase obtained from the population response of any single lattice is far from a unique specification of rat position. Assuming no additional dMEC functionality beyond the experimentally observed modulo responses, summarized above, and without reference to any specific upstream readout or decoding scheme, we ask whether the set of phases $\vec{\chi} = (\chi_1(x), \chi_2(x), \ldots \chi_N(x)) = (x \bmod \lambda_1, x \bmod \lambda_2, \ldots x \bmod \lambda_N)$ is sufficient for uniquely specifying rat position x and over what range.

Capacity of idealized modular position code
Let us first consider an idealized situation, in which position x and the lattice periods λ_α are restricted to be whole numbers and the phases $\chi_\alpha(x)$ are known without uncertainty. We will later relax these assumptions. In this case, the properties of representing a number by its modulo remainders are analytically known. The Chinese remainder theorem (CRT) can be used to prove that a modular representation with a relatively small number of moduli (lattice periods) uniquely specifies the input number (position) over a large range (Burak and Fiete, 2006; Gorchetchnikov and Grossberg, 2007).

As an example, consider a simple system with two moduli or periods, $\lambda_1 = 3$ and $\lambda_2 = 4$. As x increases progressively from 0,

$\chi_1(x)$ repeatedly cycles through the values $\{0, 1, 2\}$, whereas $\chi_2(x)$ cycles through $\{0, 1, 2, 3\}$. Starting at $x = 0$, where $(\chi_1(x), \chi_2(x)) = (0, 0)$, the values of the two phases taken as a pair do not repeat until $x = 12$, where $(\chi_1, \chi_2) = (0, 0)$ again. This simple system with $N = 2$ can represent 12 distinct positions.

In fact for any number N of lattices, the CRT guarantees that the modulo phases uniquely represent any position in the range from 0 up to

$$D = \text{LCM } \{\lambda_\alpha | \alpha = 1...N\} - 1 = (\prod_{\alpha=1}^{N} \lambda_\alpha) - 1, \quad (2)$$

where LCM refers to the least common multiple. The second equality holds if the lattice periods λ_α are co-prime, i.e., they share no common factors. For the set of five moduli $\vec{\lambda} = (13, 15, 16, 17, 19)$, any number in $0 - 1,007,759$ is uniquely specified by its residues or phases, with 1,000,000, for instance, represented by the phases $\vec{\chi} = (1, 10, 0, 9, 11)$. As in the base-10 numeral system, where a number as large as 1,000,000 can be represented by just six registers, here five moduli are sufficient to represent a similar range.

If, as in dMEC, all the lattice periods are of similar magnitude ($\lambda_\alpha \approx \lambda$), the total system capacity according to Equation 2 scales as λ^N. Thus, capacity scales combinatorially in the number of lattices, and the efficiency of encoding a number (position) as a set of phases or modulo remainders is as high as storing it in any base-b positional numeral system, where the size of the largest encodable number ($\sim b^N$) also grows exponentially with the number N of registers.

Capacity of grid cells
In reality, lattice periods and rat position are not dimensionless integers: they are real-valued quantities for which there is no notion of LCM or co-primeness. Thus, we generalize to the case in which the lattice periods λ_α, rat position x, and lattice phases are arbitrary real numbers (we continue to consider rat displacements in one dimension). In addition, we assume that the width of the firing fields at each lattice vertex and the stochasticity inherent in neural firing contribute to uncertainty in the exact phase, so that the firing phase of the dMEC population is only known to finite precision. Results on capacity based on modular arithmetic and the Chinese remainder theorem no longer apply under these conditions. In this more realistic case, what is the capacity of grid cells, and how does it scale?

The exact real-valued phases are still denoted by $\chi_\alpha(x) = x$ mod λ_α. Because the responses of neurons with larger lattice periods resemble uniformly stretched versions of the smaller period lattices (Hafting et al., 2005), the width of the neural firing fields and thus the phase uncertainty $\delta\chi_\alpha$ is a constant fraction of the lattice period λ_α for all lattices. We denote the period-independent scaled phase uncertainty, which is attributable to the width of the neural firing fields at each vertex and is between 0 and 1, by $\delta\phi \equiv \delta\chi_\alpha/\lambda_\alpha$. This error represents the intrinsic error in the ability of an ideal observer to readout the actual grid phase $\vec{\chi}$ and should not be confused with errors attributable to path integration inaccuracies, which produce a mismatch between x, the rat's internal estimate of its position, and its true position.

If $\vec{\lambda} = (6.3, 11.9, 15.4)$ cm, the phases do not exactly repeat anywhere in $(0, 2356.2)$ cm, which might naively be thought of as the capacity of the system. However, if the phases are known only up to an uncertainty of 1 cm, the range of distances that can be unambiguously represented is actually much smaller: after an excursion of 108 cm, the phases are $(0.9, 0.9, 0.2)$ cm, indistin-

guishable from their values at the origin. Thus, phase uncertainty can reduce the capacity precipitously, but it remains unclear whether capacity with phase uncertainties is combinatorial.

By analogy with the integer case, we may estimate the capacity with phase uncertainties $\delta\phi$ as follows. Because each lattice has approximately $1/\delta\phi$ distinguishable states, the effective number of states for all N lattices is $(1/\delta\phi)^N$. The code is fully combinatorial if the phases visit all these possible states while position is continuously incremented, before returning to a previously visited state. The step size to transition from one phase state to another is approximately $\lambda\delta\phi$ (if all lattice spacings are of the same order of magnitude). This counting argument suggests an upper bound for capacity,

$$D_{\text{est}} \sim \lambda \left(\frac{1}{\delta\phi}\right)^{N-1}, \quad (3)$$

which is only a crude estimate because the phase uncertainties were not treated in a precise manner. Nevertheless, Equation 3 provides a useful benchmark for comparison with the more rigorous results that follow. It indicates that, in a combinatorial code, the capacity should increase algebraically with decreasing phase uncertainty $\delta\phi$ and should increase exponentially with the number of lattices N.

To determine coding capacity precisely, we define the following distance measure d between the phase vectors $\vec{\chi}$ and $\vec{\chi}'$:

$$d(\vec{\chi}, \vec{\chi}') =$$

$$\max_\alpha \left(\min\left\{ \frac{1}{\lambda_\alpha}|\chi_\alpha - \chi_\alpha'|, \ 1 - \frac{1}{\lambda_\alpha}|\chi_\alpha - \chi_\alpha'| \right\} \right). \quad (4)$$

So long as at least one component of the phases differs by more than the phase uncertainty $\delta\phi$, the distance $d > \delta\phi$, and the corresponding positions can be differentiated by the phase code. Conversely, if all the components in the difference are smaller than the phase uncertainty, then $d < \delta\phi$ and the phases and associated positions are not distinguishable. The capacity D, or the uniquely representable range, is the smallest distance between any two positions x, x' for which the corresponding phases $\vec{\chi}$ and $\vec{\chi}'$ obey $d(\vec{\chi}$ and $\vec{\chi}') < \delta\phi$.

In contrast to the integer case, there is no analytic expression for how D depends on the set of lattice spacings and on $\delta\phi$. We therefore evaluate D numerically. The formula for d only involves the difference between the two phases: $d(\chi, \chi') = d(0, |\chi' - \chi|)$. Thus, without loss of generality, it is sufficient to begin at $x = 0$ and increase x systematically until the first x^* for which the phase reenters within a cube of side $2\delta\phi$ centered at the phase $\vec{0}$, so that $d(\vec{0}$ and $\vec{\chi}) < \delta\phi$.

To obtain the functional dependence of capacity on $\delta\phi$, we vary the phase resolution (keeping the number of lattices fixed, at $N = 12$) and numerically compute D as above. We find (Fig. 2) that capacity grows with increasing phase resolution $1/\delta\phi$ as

$$D(\delta\phi) \sim \left(\frac{1}{\delta\phi}\right)^{N_{\text{eff}} - 1}, \quad (5)$$

where $N_{\text{eff}} \approx 10.7$ is obtained by fitting the capacity curve in the example of Figure 2. Thus, capacity in the realistic case scales similarly as in the idealized case, growing as a power of the effective number of distinguishable states per lattice, roughly estimated by the inverse phase uncertainty, $1/\delta\phi$. However, the power N_{eff} is slightly smaller than $N = 12$, the actual number of
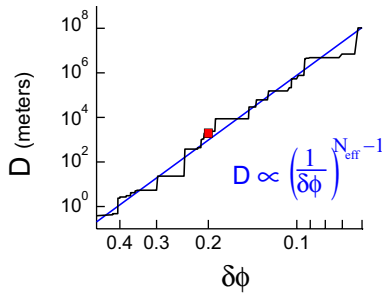
Figure 2. Capacity grows as a power of the phase resolution within lattices. The maximum uniquely representable distance grows as a power of the phase resolution, $D \propto (1/\delta\phi)^{N_{eff} - 1}$. Red dot, Phase uncertainty of $\delta\phi = 1/5$. ($N = 12$, with first lattice period = 30 cm, and increments of 4 cm per subsequent lattice.) The fit parameter $N_{eff} \simeq 10.7$, smaller than $N = 12$, but of similar magnitude.

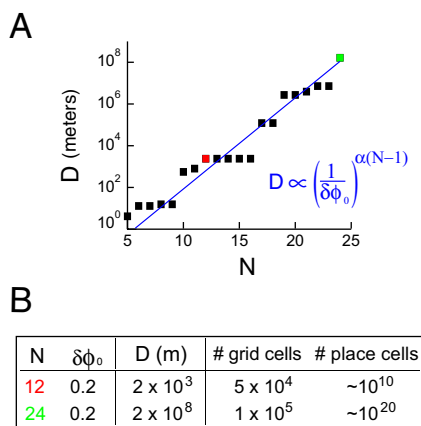| N	$\delta\phi_0$	D (m)	# grid cells	# place cells
12	0.2	2 x 10³	5 × 10⁴	~10¹⁰
24	0.2	2 x 10⁸	1 × 10⁵	~10²⁰

Figure 3. Capacity grows exponentially with the number of lattices. **A**, The maximum uniquely representable distance, D, grows exponentially with the number of lattices, $D \propto (1/\delta\phi_0)\alpha^{(N - 1)}$. For all lattices, the phase resolution is $\delta\phi_0 = 1/5$; as in Figure 2, the first lattice period is 30 cm, with 4 cm increments per subsequent lattice. The fit parameter is $\alpha \simeq 0.62$. Red dot, $N = 12$. Green dot, $N = 24$. **B**, Twelve lattices with uniformly spaced periods from 30 to 74 cm with $\delta\phi = 0.2$ can unambiguously represent an $\sim 2 \times 2$ km area with a 6×6 cm resolution. If 5000 neurons build each lattice, that would require $\sim 5 \times 10^4$ neurons. To cover the area with sparse unimodal place cell-like encoding [with 10 neurons per (6 cm)² block] would require $\sim 10^{10}$ neurons compared with the estimated 10^5 neurons in rat dMEC (Amaral et al., 1990; Mulders et al., 1997), which could unimodally represent at most $\sim 6 \times 6$ m with a (6 cm)² resolution. With 24 lattices, $D \sim 2 \times 10^5$ km (with a 6 cm resolution) in each direction, hugely in excess of the representational requirements of rats.

lattices: this modest modification is the price paid for uncertainties in phase and real, not co-prime integer, lattice periods.

Similarly, to determine how capacity scales with N, we vary the number of lattices (keeping phase resolution fixed). We find that capacity grows exponentially with the number of lattices N (Fig. 3), scaling as

$$D(N) \sim \left(\frac{1}{\delta\phi_0}\right)^{\alpha(N-1)}, \qquad (6)$$

where $\alpha = 0.62$ is a fit for the parameters of Figure 3. Therefore, despite uncertainties in phase, the scaling of capacity with number of lattices is still combinatorially large and comparable with the idealized case, at the cost of a prefactor $\alpha < 1$ in the exponent. Nevertheless, for generic choices of the real-valued lattice periods, α is of order unity.

With conservative values for the parameters ($N = 12$ dMEC lattices, periods uniformly spaced between 30 and 74 cm, and phase resolution of one-fifth of the lattice period, $\delta\phi = 1/5$), we

obtain a range for unique position representation of 2000 m per linear dimension, with a resolution of 6 cm (Fig. 3). This value is on the correct scale to cover the behavioral range of rat excursions; by including a few more lattices or assuming a somewhat better phase resolution, it is easy to obtain a nearly astronomical range of position representation [e.g., $(2 \times 10^8$ m$)^2$], vastly in excess of the rat's behavioral range (Fig. 3).

Even if we assume a high encoding redundancy, with 1000 neurons active per phase and per lattice and with the conservative set of parameters used in the preceding paragraph, the dMEC phase code would require only $\sim 5 \times 10^4$ neurons to cover a $(2000$ m$)^2$ area with a resolution of 6 cm per side. [The entorhinal cortex in rats contains $\approx 10^5$ neurons (Amaral et al., 1990; Mulders et al., 1997), which provides an upper bound for the total number of neurons in dMEC.]

Let us briefly reconsider dMEC information content from the point of view of single neurons. If a neuron fires in a lattice with phase resolution $2\pi/5$, the rat can be localized to within $\approx 1/5$ of the area; assuming 1000 active neurons per phase and per lattice as above (i.e., assuming that each lattice comprises 5000 neurons) yields a minuscule estimate of 0.001–0.003 bits per dMEC neuron. Thus, the capacity of the single neuron code or even the full population code in a single lattice is shockingly small, reminiscent of the low capacity for dense distributed codes in associative networks (Foldiak, 2002). However, when neurons are added to form a new lattice, they enlarge the overall capacity exponentially (Fig. 3) rather than algebraically, as they would for neurons added to an existing lattice population (Fig. 2). Total dMEC capacity is therefore combinatorially large, produced by pooling a number of independent, low information content networks into a structured population-of-populations or "meta-population" code.

Comparison with place cell coding. We have seen (Fig. 3) that with $\sim 5 \times 10^4$ neurons, the modulo code in dMEC can easily cover 2000 m with 6 cm resolution per linear dimension. This compares favorably with the rat's behavioral foraging range, estimated at 100 m to 1 km per linear dimension (Recht, 1988; Miller and Clesceri, 2002; Russell et al., 2005). If dMEC were instead to encode position-like idealized place cells (strictly unimodal grandmother-cell representation of position), it would require $\approx 10^{10}$ neurons to cover the same dynamic range, even if we assumed a very low redundancy of 10 neurons active per (6 cm)². The rat hippocampus contains only $\approx 10^6$ neurons (Amaral et al., 1990). With these neurons and an idealized place cell code, the hippocampus would be able to cover a range of at most 20×20 m, even with nearly no redundancy. The contrast in capacity between the idealized place cell and modulo residue codes is even more dramatic if entorhinal cortex is assumed to contain 24 lattices instead of 12 (Fig. 3). Even with quite conservative estimates, therefore, the phase code for position is vastly more efficient (as quantified by the ratio total capacity/number of cells) than an idealized place cell code.

The large capacity of dMEC has many potential uses. The large range of the dMEC code together with the regular position-based firing of grid cells suggests that dMEC forms the primary representation of position in the rat brain and can provide unique modulo phase labels for every possible location in the large foraging range of rats (corresponding to scales much greater than the periods of individual lattices). Furthermore, the capacity is large enough for the representation to have high resolution at each location. These properties together suggest, independently of dynamical considerations or network modeling (Burak and Fiete, 2006; Fuhs and Touretzky, 2006; McNaughton et al., 2006;

Burgess et al., 2007), that dMEC could be the elusive neural integrator of idiothetic rat motion cues (Hafting et al., 2005). Excess capacity may be used for error correction (discussed below) in the representation of position or to build multiple independent representations of space based on context ["multiple cognitive maps" (McNaughton et al., 1996; Touretzky and Redish, 1996; Redish, 2001)] (discussed below).

Beyond capacity

Beyond high capacity, the modulo code has several useful features and implementational advantages for position representation and updating, as described next. For simplicity, we will frequently use examples with lattice periods treated as integers, but the properties described below apply directly to real-valued lattice periods as well.

In parts of this section, we compare the modulo code with positional base-b systems. In light of the structure of the dMEC code for position and its correspondence with the RNS, it is not outrageous to imagine that the brain could instead have used a base-b positional numeral system to represent the numeric quantity of position [in the positional base-b system, the pth register represents $(x \bmod \lambda_p)$ with resolution b^p, where $\lambda_p = b^{p+1}$]. Therefore, it is possible to debate the relative merits of the two representation schemes.

Narrow range of grid periods: similar lattices without capacity cost
Modular representations allow great flexibility in the choice of moduli or lattice periods: the periods may span a geometrically large range, as in positional base-b systems (Fig. 4A, second row). In the base-b systems, the ratio of the maximum to minimum register sizes must equal the dynamic range (range over resolution) of the represented quantity. For a number represented by six registers (digits) in the decimal system, the largest register represents the hundreds of thousands scale, whereas the smallest represents ones. In the modular system, however, it is possible for all the moduli to be of the same magnitude, without sacrificing capacity (Fig. 4A, third row).

The sizes of the different registers also dictate how rapidly those registers change as the represented quantity is varied: the Nth register in a base-b scheme increments b^{N-1} times more slowly than the first register, whereas in the modulo scheme with approximately equally sized moduli, all registers increment at approximately the same rate (Fig. 4B).

What is the biological relevance of this observation? To generate lattices that span a very large range of sizes (e.g., several decades, as might be the dynamic range of position representation, estimated from dMEC capacity) requires the existence of parameters in the underlying network that span a similarly large range, a difficult task. Instead, the modular system may have an arbitrarily large dynamic range (with a suitable number of lattices), with a narrow dynamic range in the parameters of the dMEC network. This property allows dMEC to maintain a high capacity with periods that are approximately within a factor of 10 of each other in size.

A corollary of the narrow range of periods allowed in the modular representation is that the different lattices share information about rat position "democratically." In the base-b numeral systems, removal of the leftmost lattices would leave information about small rat displacements intact but obliterate knowledge of the large-scale position of the rat, whereas removal of the rightmost set would have the converse effect. In the modular system, information from all scales is present in every lattice (Fig. 4B). Lesioning a few dMEC lattices beyond the number that
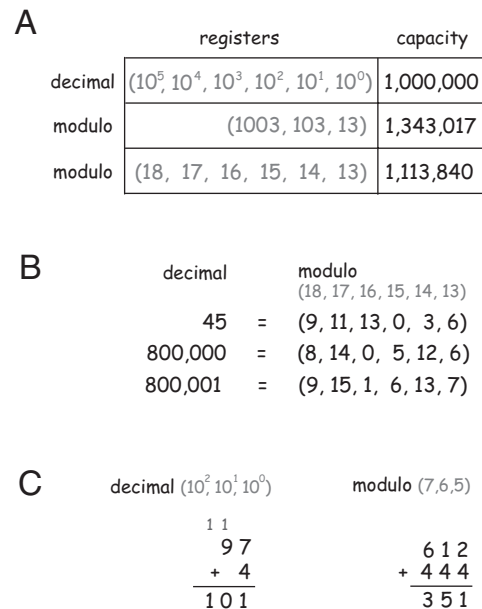

Figure 4. Neurobiologically useful properties of the modulo code. *A*, Similarly sized registers. Six registers can represent numbers in the range 0 to $(10^6 - 1)$ in the decimal fixed-base positional numeral system. However, the leftmost register represents quantities that are one million times larger than the rightmost register, and increments one million times more slowly as the represented quantity is varied. Registers in the modulo code may also span a large range (middle row) but, importantly, may be chosen to be similar in size (bottom row). *B*, Similar update rates. With similarly sized moduli (shown in gray), all registers are equally important for representing position at all scales (compare the representations of 45 and 800,000), and all registers increment at similar rates as position is varied (compare the representations of 800,000 and 800,001). *C*, Parallel, carry-free position updating. Left, Summation of 97 with 4 entails a carry operation when the 1's register wraps around in the decimal fixed-base numeral system. Right, In the modular phase representation with moduli (7, 6, 5), the same numbers [$97 \equiv (6, 1, 2)$ and $4 \equiv (4, 4, 4)$] sum to $101 \equiv (3, 5, 1)$. The register corresponding to the modulus 5 wraps around because $2 + 4 \bmod 5 = 6 \bmod 5 = 1$, as does the register corresponding to 7, because $6 + 4 \bmod 7 = 10 \bmod 7 = 3$. However, no information is carried to other registers to produce the correct result. (The examples above use integer moduli, but the principle holds for reals.)

provides redundant coding of position (see below) should degrade navigational behavior equally at all scales, big and small, larger than the individual lattice periods.

Carry-free arithmetic: independent position updating across lattices
In the modular representation, operations such as addition, subtraction, and multiplication are completely parallelized: if there is a wraparound in one of the registers, no information is passed to other registers to obtain the correct sum (Fig. 4B). This is because, if $a' = a \bmod \lambda_\alpha$ and $b' = b \bmod \lambda_\alpha$, then $a' + b' = (a + b) \bmod \lambda_\alpha$, or, in other words, the sum of two numbers is simply the modulo sum of their phases, computed separately for each register or lattice. (In contrast, in the fixed-base numeral systems, whenever one register wraps around, the next register must be incremented by an appropriate amount by carrying the spillover.)

This property of modular arithmetic has long been appreciated and exploited in computer science. Computer hardware designed to operate on the RNS instead of binary has been used to produce large speed boosts by avoiding the overhead of carrying numbers in applications such as signal processing, cryptology, and RSA, which involve repeated addition and multiplication steps (Soderstrand et al., 1986; Mohan, 2002; Bajard and Imbert, 2004).

Why is this seemingly abstract feature of modular arithmetic relevant for position representation in neurobiology? As the rat moves from x to a new location $x + \Delta x$, the mental representation of its position must be updated by addition of the displacement vector Δx. Because of the carry-free property of modular arithmetic, each lattice can directly (in parallel) use feedforward motion inputs to add Δx to its previous phase to obtain the correct phase for the new rat position. There is no need to sum spillovers from, or pass spillovers to, other lattices, even when a phase wraps. Although positional information is distributed across lattices, no coordination (and therefore no recurrent processing) is required between lattices to update position.

The carry-free property of the dMEC code shows that the dMEC representation is well suited for summing incremental movement signals to update its state, again suggesting, independently of any dynamical considerations or modeling (Burak and Fiete, 2006; Fuhs and Touretzky, 2006; McNaughton et al., 2006; Burgess et al., 2007), that dMEC might be the locus of idiothetic path integration in the brain (Hafting et al., 2005).

Distance metric is not preserved for metric comparisons
If the rat moves by an amount much smaller than the period of the smallest lattice ($\lambda_{\min}$), the updated phase vector is close to the original phase vector. For instance, if the rat started at position $x = 0$ with phases $(0, 0, 0)$ for a set of lattices of periods $(17, 18, 19)$, the metricity of real space is preserved in the phases for excursions smaller than 16 steps. Thus, the grid cell representation can be directly and easily used over these scales to make metric comparisons of distance.

Over large distance scales, metric information is not lost in the modular dMEC code, because the mapping from position to phases is unique and the map has an exact inverse, but making real-space distance comparisons is difficult. This is because far-away points are regularly mapped very close to each other in the grid cell representation. In the example above, the nearby phases $(16, 16, 16)$ and $(16, 17, 18)$ represent $x = 16$ and $x = 5813$, which are separated by 5797 steps, nearly the full range of the representation. Figure 5 illustrates that points in the space of modulo phases do not share the same metric as points in real space (rat location). The cross-correlation of the dMEC population vectors corresponding to locations $x, x' = x + \Delta x$ decays monotonically with Δx over short distances (Δx smaller than the blob width in the largest lattice), but, over larger distances, the correlation varies non-monotonically with separation Δx. Thus, beyond the width of a blob, closeness in real space is unrelated to closeness in the grid cell representation, and a simple linear readout (or even a linear summation followed by a monotonic nonlinearity) of dMEC cannot compute distance in real space.

This observation prompts the question of whether the brain translates the grid cell representation into a form that is more amenable to explicit metric computations (and, if so, how) or whether it uses the phases of entorhinal cortex to perform other navigation-related tasks that do not require explicit inversion and metric comparisons. We consider this question below (see below, Downstream uses of the grid cell code).

Robustness
In principle, the potentially vast excess capacity of dMEC (Figs. 2, 3) can be used to redundantly represent the same information so that faulty phases may be corrected. In the context of error correction, the fact that small perturbations in phase correspond to large differences in the represented position is an asset. Let us consider one, not necessarily optimal, example of error correc-
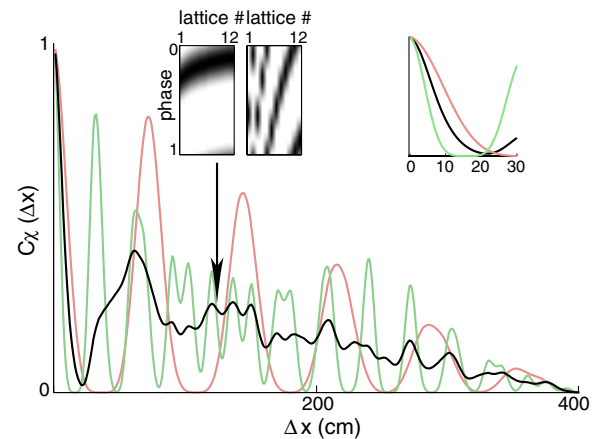

Figure 5. Decorrelation of dMEC population response with distance. Black line, Cross-correlation between the full dMEC vectors (representing activity in all phases, all lattices) corresponding to pairs of locations separated by distance Δx. Green (pink) curves show cross-correlations for the dMEC phase vector of only the two smallest (largest) lattices. The dMEC population correlation is high and monotonically decreasing with Δx for small separations (central peak, magnified in rightmost inset): in this regime, distances in the space of dMEC responses are proportional to distances in real space. However, the dMEC population vector has completely decorrelated for Δx, comparable with the width of a blob in the largest lattice [full-width at half-maximum (FWHM) of ~18 cm; the central peak width in the green (pink) curve is approximately the blob width in the smallest (largest) lattice], beyond which dMEC correlations begin to vary nonmonotonically. Middle insets, Two-sample dMEC population vectors, representing two positions separated by $\Delta x = 140$ cm, to illustrate the degree of decorrelation in the population activity. The abscissa shows the index of different lattices, arranged in order of increasing period. The ordinate shows the phase of the active population within the corresponding lattice. Grayscale intensity represents the analog level of activity (black is maximally active). Middle left, The population vector at $x = 80$ cm; middle right, the population vector at $x = 220$ cm. Parameters: $N = 12$ 1-D lattices, periods uniformly spaced in (30,74) cm. The binary population vector χ ($N \times M$ dimensions) has $\chi(\alpha, i) = 1$ if $i =$ ceil[M (x mod λ_α)/λ_α] and 0 otherwise. $M = 100$ is the number of different phase bins per lattice. The population activity vector $\vec{r} = \{r_{\alpha,i}\}$ is the convolution within each lattice of the phase vector with a Hanning kernel, so that, although the activity of the α, ith phase group peaks at $\phi_{\alpha,i}$, it also responds to neighboring phases, with FWHM of 1/4 of the total lattice period.

tion [see other examples by Soderstrand et al. (1986) and Mohan (2002)]. For lattice periods $(15, 17, 19, 22)$, the unique representable range is $[0, 62985]$. If the rat never ventures outside the range $[0, 250]$, position can be represented by any two of the four lattices, rendering the other two redundant. In the modular representation, $200 = (5, 13, 10, 2)$. An error of ± 1 in any one of these phases corresponds to positions >6000. Because any of these perturbations to the phase vector represents positions far outside the range of interest, an error is easily flagged even without knowing which phase is wrong. Ignoring one of the correct lattice phases does not produce candidate positions in the range of interest; ignoring the incorrect phase does produce a candidate position in the relevant range, which is the true position. In short, dropping phases typically produces exact error correction (if the original error was in the form of small perturbations to a few phases), even without knowledge of the correct phase vector or the identity of the wrong phase.

Downstream uses of the grid cell code
We emphasize that, until now, we have simply characterized or mathematically elucidated the properties of the grid cell code for position. Therefore, our analysis has been model-free. Next, we consider possible ways in which the grid cell code for position may be read out by downstream areas. What follows is necessarily more speculative.

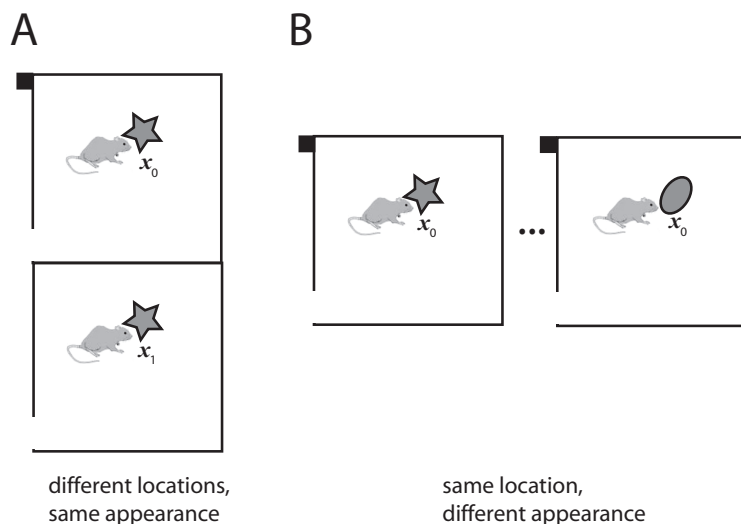Let us consider a concrete example: suppose that, for the set of

maintaining a fully combinatorial capacity because each place name is based predominantly on the identity of active neurons (whether a neuron is active or not), not their rates. Thus, the dMEC code, although itself dense, nevertheless facilitates the formation of a sparse place-label readout (like place cells in hippocampus), in a way that dense graded codes cannot.

The place-label readout can help with certain navigation tasks and not others. Because place labels do not extract the metric information present in the dMEC phases about the rat's location, they cannot be used to compute and represent vectors for straight-line homing to a location after random excursions from it, over large ranges and in general environments. Nevertheless, place labels can be instrumental in navigation, as we describe below.

If grid cell phases are at least partly based on integrated rat movement signals, whether computed within dMEC or elsewhere, the phase labels would enable the rat to know that it has arrived at a landmark even when the sensory marker for that landmark is absent or its appearance has changed (Fig. 6B) because the grid cell phase labels, computed by summing displacements as the rat arrives at the location from home, will be the same. Similarly, the rat can discriminate between two different but similar-looking locations (Kuipers, 2000) (Fig. 6A) because the grid cell phase labels for the two locations will be different. Thus, although place labeling constitutes a non-metric readout of the grid cell code, its most useful properties from the point of view of navigation actually arise if the labels are derived and updated using strict metric rules for movement-based integration.

Figure 6. Uses of the grid cell code: nonmetric place-label readout. The black square represents home base or the rat's starting location. *A*, Metrically updated place labels that themselves convey no metric information can help to differentiate separate but similar looking locations (x_0 vs x_1). *B*, A landmark may change appearance over time; however, the metric updating of dMEC phases as the rat moves from a reference point to the landmark generates an invariant dMEC phase vector at the landmark. The consistent dMEC state helps to correctly identify a location despite the inconsistency of sensory cues.

lattice periods (15, 17, 19, 22), the phase vector at some perceived location x^* reads $\vec{\chi}^* = (5, 13, 10, 2)$. The readout may treat this phase vector as a unique neural label or name for x^* (unique over a range of 62,985 units), without using the fact that $\vec{\chi}^*$ corresponds to a displacement of 200 units from $\vec{\chi} = \vec{0}$. Alternatively, the readout may actually perform the inverse mapping from modulo phases to displacements, to obtain and exploit the metric information inherent in the modulo code.

We refer to these two complementary, and not necessarily mutually exclusive, readout possibilities as (1) a place-label readout and (2) an explicitly metric readout and discuss each below. We pause briefly here to stress that we are only considering how the grid cell phases $\vec{\chi}$, which represent x, the rat's estimate of its location, are read out and used downstream in the brain and not whether x is an accurate estimate of the actual rat position. Admittedly, the utility of any readout of x depends on whether x is itself a reasonably good estimate of the rat's true position over the relevant scales; purely idiothetic path integration may produce large errors in position estimation. However, what is considered a sufficiently accurate estimate to be useful may depend on the task. In what follows, we assume that the dMEC position estimate is arrived at based on a combination of path integration computations and external sensory cues (landmarks), which can correct gross errors in position estimation (Etienne et al., 2000).

Place-label readout
The position code in dMEC is distributed and dense. Distributed dense codes can have a high capacity in the sense that they can represent a large number of variables with relatively few neurons. However, a well known problem with distributed dense codes, made much worse if the code consists of graded activity patterns, is that they are difficult to read out by neural networks, because different code words or labels can have large overlaps with each other (Foldiak, 2002). For instance, if the code words consisted of the analog firing patterns of a neural population (e.g., a rate-based code), all neurons would fire at most locations, with the primary difference between different words being the graded levels of activity. The dMEC code ameliorates this problem while

Place cells in the hippocampus may be viewed as the sparse readouts of the unique but distributed place labels given to locations by dMEC. In fact, recent modeling studies have shown how sparse place cells might be constructed from dMEC inputs, using appropriate feedforward weights between dMEC and the hippocampus [by summing input from grid cells with different periods and phases chosen so that the grid cell responses have peaks at the place field center (O'Keefe and Burgess, 2005; Rolls et al., 2006; Solstad et al., 2006; Franzius et al., 2007)]. This observation suggests that dMEC place labels are useful in navigation for many of the same reasons as have been widely proposed for place cells; for instance, the sparsened place cell readouts of the dMEC place labels can, together with the help of associative learning, form memories that chain together sequentially visited locations to aide in the learning of specific landmark-based routes between locations (Blum and Abbott, 1996; Redish and Touretzky, 1998; Foster et al., 2000). Additionally, if dMEC is the path integrator and if the connectivity from dMEC to place cells and back is symmetric, then place cells in the hippocampus, if anchored to landmarks through sensory inputs, can be used to reset the dMEC phases and correct errors in the internal estimate of position (Redish and Touretzky, 1997; Samsonovich and McNaughton, 1997; Redish, 2001; O'Keefe and Burgess, 2005).

In the typically small areas (long 1-D tracks or ≈1 m diameter enclosures) used in experiments, hippocampal neurons tend to form unimodal place fields that resolve position with relatively

high accuracy and appear to cover the space roughly uniformly (O'Keefe and Dostrovsky, 1971; Wilson and McNaughton, 1993). Above (see Capacity), we concluded that hippocampal capacity rules out the possibility that large enclosures could be fully remapped with high resolution [$\sim(10 \text{ cm})^2$] using idealized unimodal place cells. Thus, we predict that in large enclosures, sparse location labels in the form of unimodal place cell firing cannot be assigned uniformly. Instead, place cells that continue to fire unimodally in large enclosures must disengage from uniformly covering all space with narrow place fields and only represent select locations (e.g., edges, landmarks, rewarded locations, etc.) with high resolution.

An alternative possibility is that hippocampal place cells will display multimodal firing fields in larger enclosures. Indeed, some place-sensitive cells in hippocampus fire at multiple locations in an enclosure (McNaughton et al., 1983; Muller et al., 1987). If multimodal responses are the norm in large enclosures, the hippocampal code for place is actually a (sparse) combinatorial code rather than a grandmother-cell code, and cells with multimodal responses cannot properly be called place cells. The resolution of this issue remains an important topic for experiments to address. Our capacity statement is not in contradiction to this alternative; we simply note that, if place cells retain their hitherto observed unimodal firing characteristic in large enclosures, they cannot match (or even approach) the range and resolution of positional information in dMEC.

The prediction of selective, highly non-uniform assignments of place fields in larger enclosures (if they remain unimodal) is qualitatively different than would be expected if place cells were the primary general-purpose encoders of rat position and if they were the loci of neural path integration, as suggested by some models (Samsonovich and McNaughton, 1997; Tsodyks, 1999), because path integration relies critically on translation invariance in the representation of space. Indeed, consistent with our prediction, several experiments show that place field coverage can be highly non-uniform (Eichenbaum et al., 1989; O'Keefe and Burgess, 1996; Hartley et al., 2000; Hollup et al., 2001).

Because the primary role of dMEC in the place-label scenario is to label locations, the absolute phase values are crucial (in contrast to relative phase differences between positions, as in the explicitly metric readout scenario described below). Thus, if the rat arrives at a landmark that it recognizes from sensory inputs as familiar but by starting from an entirely novel location without landmarks along the way, so that its phase vector on arrival is not the same as its phase vector at previous visits to that landmark, the dMEC state should abruptly reset to the previously assigned phases.

Another reason why absolute phases may be important is if the dMEC code signals context in addition to position. dMEC may represent non-metric quantities such as context but also retain a purely metric representation of space locally (over large spaces) by breaking up its vast capacity of states into contiguous blocks, with each block representing space metrically but different blocks representing that space in different contexts ("multiple cognitive maps").

A pair of phase vectors corresponding to a massive difference in the represented position (e.g., by an amount larger than a rat's range) could be used to represent the same point in space in different contexts. Within this picture, a context-dependent remapping of space corresponds to a differential shift in the phase across lattices. For hippocampal place cells, generated by summing several dMEC inputs, this dMEC remapping should produce a visually dramatic transformation, with fields appearing in
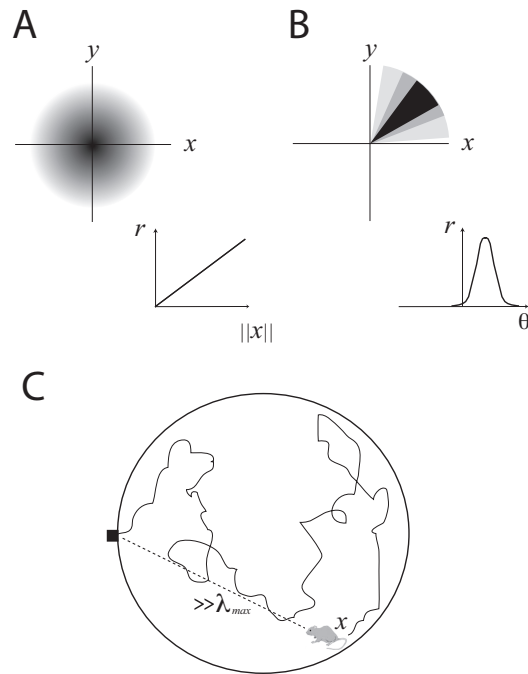

Figure 7. Uses of the grid cell code: explicit metric readout. **A**, An example of an explicit metric readout of the dMEC phase code: if the firing rate of a readout cell is proportional to displacements of the rat from a starting location, small changes in the neural response correspond to small changes in rat location and vice versa and constitute an explicitly metric code for displacement. Top, 2-D; bottom, 1-D. **B**, An example metric readout for angular displacements: the firing of a cell is proportional to closeness to a reference orientation. Top, 2-D; bottom, 1-D. **C**, Explicitly metric readouts may be used to compute the straight-line path home after random excursions over long distances. The black square represents home base.

new locations and disappearing from old locations, or with cells that were previously inactive in an area becoming active, and vice versa.

Explicit metric readout
For a rat to perform navigation tasks such as estimating the direction and straight-line distance toward home following random foraging trajectories (Fig. 7C), it needs to (1) estimate its current position from its displacements and any available sensory cues, and (2) compute a homing vector that points from its current position to home, from which the direction and distance to home can be read out. Even if a rat is incapable of computing an accurate straight-line homing vector over large scales, it is likely that the animal must have a vague sense of the direction back home; similarly, it must be able to approximately gauge the relative straight-line distances from home to two different and distant food sources (e.g., which one is closer to home), even if it took a circuitous trajectory to each one. Because dMEC phases encode the coordinates of the rat's current location, dMEC already reflects the results of computation 1 above (whether the computation to estimate the rat's position is done by dMEC or an upstream area). Additionally, because dMEC phases are always relative to a starting phase, dMEC also encodes the vector displacement of the rat relative to the reference location, information necessary for performing computation 2.

However, metric information about rat displacement contained in the dMEC code is difficult to extract. This is because, as noted earlier, it is not possible for a single-layer neural network readout to reconstruct metric quantities relating to the rat's position in real space from the dMEC code, for general values of the rat's possible location. Therefore, we expect that converting the

dMEC code into an explicitly metric representation useful for homing requires a dedicated processing network. By explicit metric readout, we mean the transformation of the dMEC code into another representation that represents (through a continuous mapping from space to neural representation and vice versa) metric properties of space in a way that distance or angular comparisons can be made without a lookup table, for instance, one in which cells fire whenever the rat is at a particular angle (or distance) from the home base (Fig. 7*A,B*).

It is possible for a neural network-like readout to convert modulo phases into an explicitly metric representation, as described in Appendix. The particular network in Appendix involves recurrent circuitry, but it may be possible for another network to accomplish the task using a multilayered architecture instead.

It is unknown whether behaviorally rats can compute rough homing vectors over displacements that are large compared with the lattice periods in dMEC and, thus, whether they possess the computational ability to invert the modulo phase code for position. Rats can home accurately, even without any external sensory feedback, after random foraging excursions in enclosures of ≤2 m diameter (Whishaw and Maaswinkel, 1998). However, over such short distances, homing may not require a separate explicitly metric readout, because the grid cell representation metrics resemble real-space metrics (Fig. 5).

Suppose that physiology experiments locate neurons in the rodent brain whose responses are explicitly metric functions of position over large distances and are independent of the taken path. Such neurons, if they exist, are likely to be the explicit metric readouts of the dMEC phase code. The existence of these neurons is likely to be predictive of behavioral abilities in rats to estimate the distance or angle toward home over large excursions (Fig. 7*B*).

Conversely, suppose that behavioral experiments in large enclosures find that a rat can perform roughly straight-line homing after random trajectories that take it farther than λ_{max} from home. Such behavioral abilities suggest the likely existence of neurons with an explicitly metric readout of rat position from dMEC. The potential to home over distances greater than λ_{max} based on dMEC input is a qualitatively, not quantitatively, different phenomenon than homing over short distances, because it requires an entire machinery for the construction of explicit metric readouts and is open for behavioral testing.

Finally, if the dMEC code were only used in the computation of homing vectors, differences in the grid cell phases vectors between two locations might be far more important than absolute phases. In such a situation, specific sets of phases need not be attached consistently to a set of landmarks; rather, the dMEC phase assignment could be arbitrary for encounters with the same landmark on different days and occasions, as long as phase differences between different locations were consistent along the animal's foraging path.

Experimental tests

Our only assumption, or central premise, in the analysis of the grid cell code (capacity and arithmetic properties) was that grid cells with a regular periodic response as a function of position remain periodic with unchanged periods in any enclosure, including in enclosures of size $\gg \lambda_{max}$ [e.g., $\approx (50 \text{ m})^2$] and that their response remains independent of path. Current evidence from experiments in small enclosures suggests that this assumption is correct, but it will be important to test it directly when the net displacement in rat position is large. Sampling a few activity

blobs in each of two distant neighborhoods within the large enclosure is sufficient to determine both path independence and periodicity.

Once the above assumption is made, the properties of the grid cell code follow automatically from pure mathematical principles and are therefore model independent. For example, the capacity of the code does not depend on how grid cell activity is processed in downstream areas of the brain. In other words, the characterization of the properties of the grid cell code for position neither depends on nor implies any specific model for the generation or readout of grid cell firing. Can we use this characterization of grid cell response to propose hypotheses that can be tested and falsified?

We have established that grid cells unambiguously specify rat position over a large range and that the dMEC code has notable implementational advantages for position representation and incremental updating. The primary testable hypotheses emerging from these considerations, which are complementary to approaches based on modeling network dynamics, is that dMEC must be the primary neural representation of rat position in large 2-D environments (based on capacity) and is likely to be the locus of path integration in the rat brain (based on arithmeticity and capacity). When results on the general properties of the dMEC code are combined with assumptions about how downstream areas may use dMEC inputs to perform spatial tasks, it is possible to generate more specific experimental predictions.

General predictions, independent of readout scheme
(1) The arithmetically convenient properties of the grid cell code for position suggest that dMEC is the idiothetic path integrator in the rat brain. The most basic test, not yet performed, is to determine whether grid cells reflect path-integrated responses in the absence of all spatially informative external sensory cues, i.e., does activity remain grid-like? The next is to pinpoint the locus of path integration through selective lesion of grid cells or their input fibers. Studies involving selective lesion and behavior show that entorhinal cortex is part of the path integration circuit (Alyan and McNaughton, 1999; Parron and Save, 2004) but have not probed the specific contribution of dMEC.

(2) The large range ($\gg \lambda_{max}$) of unique position representation suggests that dMEC is the primary repository of general-purpose, high-resolution and large-scale position information in the rat brain, quite independently of whether or not it performs path integration. In an experiment that is not yet technologically feasible, but could be in the future, the hypothesis that the rat uses the dMEC code as its primary reference for its sense of position could be directly tested. Shifting the phase vector by $\Delta \vec{\chi} = \Delta x$ mod ($\vec{\lambda}$) should produce a shift by Δx in where the rat thinks it is. (Where the rat thinks it is could be assayed easily by whether the rat freezes or displays other unconditioned aversive responses, if the shifted location corresponds to where the rat has received multiple foot shocks or some other aversive stimulus.) The experiment to shift the dMEC phases requires activation of a specific phase vector in dMEC. Because neurons in dMEC do not appear to be physically grouped by phase, this experiment requires tools capable of identifying and stimulating groups of functionally related neurons.

(3) The similarly sized periods of grid cell lattices imply that, in any spatially information-rich readout (metric or non-metric, but over large spaces), each readout cell must pool input from many or a majority of dMEC lattices. This is because small numbers of lattices do not uniquely specify rat position and because lattices of all periods share information about rat position equi-

tably. Indeed, neurons in the septal (dorsal) half of the long axis of the hippocampus do appear to receive inputs from across the dMEC band (Witter and Groenewegen, 1984).

(4) For capacity scaling, rats with partial lesions that spare linearly increasing numbers of grid lattices should display exponentially better performance, toward normal. This prediction can only be probed in enclosures $\gg \lambda_{max}$, because it is at this scale that information must be pooled across lattices to uniquely specify position.

Predictions specific to the place-label readout scheme
(5) A rat with a large dMEC lesion (missing most lattices) should be unable to recognize the same location if it looks different after excursions $\gg \lambda_{max}$. Such a test could involve an annular enclosure with radius $\gg \lambda_{max}$ in which the rat runs unidirectionally, with landmarks strewn along the circumference, including a specific landmark (that changes appearance or disappears between visits) at which, on the rat performing a specific behavior, a food or drink reward is given. A control rat should be able to do this task.

(6) A rat with a large dMEC lesion will be unable to disambiguate two similar-looking locations separated by a distance $\gg \lambda_{max}$, even if it is able to do so over distances comparable with a lattice period. The experimental setup could be the same as above but with one rewarded and one nonrewarded location, both of which look similar. A control rat should be able to do this task.

(7) Place cells cannot form a general-purpose, high-resolution representation of all space in enclosures of size comparable with the foraging ranges of rats (like dMEC can) with a unimodal place-like code; instead, spatially compact place fields will only form at salient locations.

(8) For multiple cognitive maps and remapping, differential shifts in the phase across different lattices in dMEC should cause global remapping of many place cells in the hippocampus. Experiments show that global place cell remapping is accompanied by phase shifts in dMEC (Fyhn et al., 2007); however, these experiments have not probed whether the phases in different lattices of the same rat shift by different amounts, as we predict.

Predictions specific to the metric readout scheme
(9) dMEC is necessary for homing over distances $\gg \lambda_{max}$, if the rat can do such homing.

(10) If the rat can home over distances $\gg \lambda_{max}$ behaviorally, there must be a separate readout area that pools inputs from dMEC and generates an explicit metric representation of position. The converse is likely to be true too: if a readout area exists that pools input from many dMEC lattices and has an explicit metric representation of space over distances $\gg \lambda_{max}$, then the rat is likely to be able to home over those scales. (An explicit metric readout is not necessary for homing over distances $\approx \lambda_{min}$.)

(11) When the enclosure is changed from a square to a circle (inscribed in the square), the period of the recorded grid cells is observed to grow slightly (Fyhn et al., 2007). If the percentage change in grid cell period is coherent across all lattices of an individual, this would correspond to a scaling of space and produce a small misestimate of position by the same amount as the scaling of the grid periods. If different lattices within one animal were to expand by different amounts, that would produce a serious problem for the metric readout mechanism. A metric readout can remain a viable possibility, only if different grids change in a consistent way.

A common theme in many of these predictions is that experiments in much larger enclosures, either in real space, aided by advanced telemetry, or in virtual space (Dombeck et al., 2007), should yield qualitatively, not just quantitatively, new insights about navigational computation and the representation of space in the rat brain.

This particular list of predictions is the product of our limited imaginations. A modulo code for position in dMEC is likely to have numerous other testable implications, to be imagined by others, for experiments involving contextual place cell remapping, readout neurons, and lesions.

Discussion
Summary
We analyzed the relationship between two simultaneously recorded quantities: rat position and dMEC activity. For this reason, we did not have to rely on models of the dynamics of response generation in grid cells or on models for the readout of the grid cell code. The characterization of the transformation from position to grid cell response followed from purely mathematical principles and allowed us to deduce the capacity and arithmetic properties governing position representation in the grid cell code. The large range of the dMEC code for position suggests that dMEC is the primary neural repository where the rat stores its estimate of position over large scales (however rough that estimate might be) and to which the rat refers for obtaining a sense of its location. The high-resolution and carry-free arithmetic property of the code hint that dMEC is the idiothetic path integrator and that it may provide fine spatial information for tasks relating shorter segments of the trajectory. Finally, the many unique properties of modular codes constrain the ways in which dMEC phases may be used by readout areas for navigation, leading to testable predictions about the properties of potential readout schemes.

The present work has considered the properties of the dMEC modular code for representing estimated position; it has not been concerned with how well the estimated position approximates the rat's true position, but ultimately, the utility of both our proposed readout schemes (and for that matter, the general utility to the rat of retaining any internal estimate of position) will hinge on the accuracy of the internal position estimate. If the error in the internal position estimate is similar to the rat's distance from home, there is little value in storing a representation of position or constructing readouts from it. However, if, with help from sensory cues, the position estimate error is smaller than the total distance from home, both readout schemes could have complementary uses for spatial navigation.

Codes in the brain
It is extraordinary that dMEC has devised an encoding scheme for position, analogous to the residue number system, that is simultaneously combinatorial in capacity, has a register structure (by virtue of positional information that is broken down and distributed across lattices in a structured meta-population code), and is amenable to parallelized arithmetic operations for incremental position updating. The capacity and register structure of the grid cell code make it comparable, for the representation of a numeric quantity (position), to the fixed-base positional numeral systems, among the most highly developed human tools for this purpose (Table 1). The carry-free arithmetic property and narrow register range of the modular code actually make it implementationally superior to fixed-base numeral systems for position representation in the neurobiological context.

To our knowledge, a numeral system neural code is unique compared with all other known examples of coding in the brain (Abbott and Sejnowski, 1999), for instance, codes that encode metric variables but are not combinatorial in capacity [e.g., rate-proportional coding for horizontal eye deflections in the oculomotor system (Lopez-Barneo et al., 1982) or unimodal population coding like the angular tuning of head direction cells (Taube et al., 1990) or the ultrasparse coding of time during song in premotor songbird neurons (Hahnloser et al., 2002)] or codes that are thought to be combinatorial [e.g., distributed odor representations in the antennal lobe of insects (Laurent et al., 2001)] but represent predominantly non-metric variables and lack apparent register structure and arithmeticity.

1-D versus 2-D analysis

For expositional and computational simplicity, our analysis was performed for a 1-D analog of the full 2-D system. How can we justify this simplification, and what do we expect in two dimensions? All properties above in Results, in the section Beyond capacity, carry over in a straightforward way regardless of the dimensionality of the represented variable, just as they carry over to real-valued numbers in one dimension. Capacity is less straightforward. In the hypothetical situation that all lattices share the same angular orientation, the rat's position in 2-D space and the 2-D phase represented by each unit cell can be decomposed into the projections along the principal lattice directions. Thus, the 2-D representation may be trivially regarded as two sets of independent 1-D phases, and our capacity analysis applies directly to each 1-D phase. More generally, if the lattices have different orientations, as is possible in dMEC, it is not trivial to decompose the 2-D phase into two 1-D phases for capacity analysis. Nevertheless, there is no reason to expect that the scaling of capacity with lattice number should be qualitatively different: a study modeling place cell responses, computed by summing dMEC inputs (Fuhs and Touretzky, 2006), shows that place cell diversity actually grows (overlaps shrink) as the diversity of relative dMEC lattice angles increases, an indirect indication that overall capacity does not decrease if different dMEC grids are unaligned.

Tradeoff between capacity and dynamics

Capacity scales very differently with neuron number in dMEC, depending on whether additional neurons are devoted to enhancing the phase resolution within each lattice (capacity grows as a power of phase resolution) (Fig. 2) versus if they are devoted to the construction of additional lattices with different periods (exponential increase in capacity) (Fig. 3). This difference could help to explain why blob size (the size of firing fields at each vertex of the lattice), which may be inversely proportional to the phase resolution, is relatively large: neurons are allocated to construct more lattices at the expense of phase resolution within each lattice, for a greater gain in capacity. However, why does the system stop at the observed phase resolution and number of lattices rather than further reducing the former and increasing the latter? Network models of the dynamics of dMEC activity suggest that the accuracy of position estimation based on path integration is already strained by the number of neurons allocated per lattice and suffers significantly if the number of cells within a lattice is reduced (Burak and Fiete, 2006, 2007) (Y. Burak and I. R. Fiete, unpublished observations). Although capacity considerations may drive neural allocation toward the formation of more lattices with fewer neurons each, dynamical constraints on the accuracy of path integration, if dMEC is the path integrator, as our results suggest, may pull neural allocation in the opposite direction. The

tradeoff between capacity and dynamical constraints may thus set some optimal value for the number of lattices and the number of neurons comprising each lattice in dMEC.

Optimality

The modular code for position is compact, but is it optimal? Capacity could be improved in at least one obvious way: by coding position with two explicitly 1-D representations, as in the Cartesian or cylindrical coordinate schemes, rather than with one 2-D representation. For grid cells, a hypothetical decomposition would consist of $2N$ 1-D lattices (N for each spatial dimension), instead of N 2-D lattices. Activity in each 1-D "lattice" would be like a slice though a principal lattice direction of the corresponding 2-D lattice. The lattices for each of the two spatial dimensions would only increment their phases for rat movements along that direction. To obtain a dynamic range r in each spatial dimension, the required number of neurons scales like r^2 for a 2-D lattice but only like $2r$ for two 1-D ones. If the representation of space were broken into two independent variables, as imagined here, then capacity would not be a great bottleneck even for sparse grandmother cell-like representations, with each cell responding unimodally around one location in one dimension. However, such a representation would be far from a place code, because the firing of each cell would represent an entire strip, not a single neighborhood, of 2-D space.

Evolution has proven itself capable of producing structures that decompose orthogonal stimulus dimensions into independent processing channels [for example, three orthogonal vestibular canals in vertebrates measure accelerations in three-dimensional space (Della Santina et al., 2005), and independent channels convey information about 2-D wind direction in cockroaches (Kolton and Camhi, 1995)]. It is unclear what constraints led the brain to represent space by one 2-D system rather than two separate 1-D ones. Possibly, a potential gain in capacity is traded for a more decorrelated representation: capacity is much greater with two 1-D representations, but the outputs remain heavily correlated across positions if rat position is varied along one dimension, even by a large amount, while keeping the other fixed (e.g., varying x but keeping y fixed if the representation is Cartesian, or varying θ while keeping r fixed in cylindrical coordinates). With the 2-D dMEC code, motion along any direction and by any amount produces approximately equal (and relatively large) amounts of decorrelation, facilitating the formation of sparse decorrelated representations of different rat positions downstream (e.g., by place cells in the hippocampus).

Grid cells and the concept of a map of space

A single dMEC lattice partitions space into a regular grid, superficially like the grids overlaid on a city map. However, crucially, grid cells do not assign unique labels (like 7-E, 3-B) to each unit cell (responses in all unit cells are identical), so there is no sense in which the grid can help to label or localize landmarks in space. When many or all dMEC lattices are considered together, they do uniquely specify locations over spaces much larger than the scale of any single grid. However, the composite code from multiple lattices is aperiodic and lacks any grid-like structure. Thus, there is no analogy between grid cell firing and grids on a city map. Furthermore, grid cell firing does not specify at any one instant in time a layout for all space. Rather, it is a system that produces the label for any location once the rat moves to that location. In an updated analogy, the grid cell population output more accurately resembles the output of a global positioning system device than a map.

Appendix

Are there plausible neural network schemes that could convert grid cell phases into explicitly metric representations of rat position? As one example, a single-layer recurrent network, like the one proposed by Sun and Yao (1994), may perform such an operation. The network takes as its inputs the grid cell phases (in one dimension), producing as its output the firing of a neuron whose rate r is equal to the displacement x of the rat, if it obeys the following dynamics:

$$\frac{dr}{dt} = -\frac{r}{\tau} - \frac{1}{C}\sum_\alpha |\hat\chi_\alpha(r) - \chi_\alpha| + \Delta, \tag{7}$$

where x is rat position; $\chi_\alpha \equiv x \bmod \lambda_\alpha$; $\hat\chi_\alpha(r) = r \bmod \lambda_\alpha$; Δ, $1/C \ll 1$; and $\tau/C \sim D/\Delta$, where D is the largest x that can be represented by the moduli.

Starting at $r = D$, r is guaranteed to reach a steady-state value $\in [x - \Delta, x + \Delta]$. The quantities $\hat\chi_\alpha(r)$ may be computed from the instantaneous value of r by other networks that take r as their input. Through numerical simulation, we find that this algorithm for remapping modulo phases to a rate-proportional code works even when the lattice periods λ_α are not co-prime or are non-integers (data not shown).

This example illustrates that plausible neural network operations can implement a conversion from modulo residues to an explicitly metric representation. Although the idiothetic path integrator must have high resolution to avoid accumulating rounding errors from successive position updates, readouts of the path integrator may in principle represent position far more coarsely. However, the readout implementation above requires a relatively high-resolution internal representation of position (as encoded in the firing rate r) for the network to converge to the correct solution. Therefore, a large number of neurons would be required to accurately encode r over the desired range. [If units (neurons) each have a fixed, finite dynamic range, i.e., if the maximum firing rate and the resolution with which firing rate changes can be determined are capped, the number of required units grows linearly with the dynamic range of the represented variable in a rate-proportional code. Thus, rate-proportional codes do not have high capacity (Table 1).] It is possible that different recurrent networks, or possibly networks with feedforward architectures, could perform a similar conversion. It is an open question whether an alternative network could do so without requiring a high-resolution internal representation of position.

References

Abbott L, Sejnowski T, eds (1999) Neural codes and distributed representations. Cambridge, MA: MIT.

Alyan S, McNaughton BL (1999) Hippocampectomized rats are capable of homing by path integration. Behav Neurosci 113:19–31.

Amaral DG, Ishizuka N, Claiborne B (1990) Neurons, numbers and the hippocampal network. Prog Brain Res 83:1–11.

Bajard J-C, Imbert L (2004) A full rns implementation of rsa. IEEE Trans Comput 53:769–774.

Barry C, Hayman R, Burgess N, Jeffery KJ (2007) Experience-dependent rescaling of entorhinal grids. Nat Neurosci 10:682–684.

Blum KI, Abbott LF (1996) A model of spatial map formation in the hippocampus of the rat. Neural Comput 8:85–93.

Burak Y, Fiete I (2006) Do we understand the emergent dynamics of grid cell activity? J Neurosci 26:9352–9354.

Burak Y, Fiete I (2007) A continuous attractor model for grid cell activity. Paper presented at the Computational and Systems Neuroscience meeting, Salt Lake City, UT, February.

Burak Y, Brookings T, Fiete I (2006) Triangular lattice neurons may implement an advanced numeral system to precisely encode rat position over large ranges. arXiv e-print server, Cornell University, arXiv:q-bio/0606005v1.

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

Della Santina CC, Potyagaylo V, Migliaccio AA, Minor LB, Carey JP (2005) Orientation of human semicircular canals measured by three-dimensional multiplanar ct reconstruction. J Assoc Res Otolaryngol 6:191–206.

Dombeck DA, Khabbaz AN, Collman F, Adelman TL, Tank DW (2007) Imaging large-scale neural activity with cellular resolution in awake, mobile mice. Neuron 56:43–57.

Eichenbaum H, Wiener SI, Shapiro ML, Cohen NJ (1989) The organization of spatial coding in the hippocampus: a study of neural ensemble activity. J Neurosci 9:2764–2775.

Etienne AS, Jeffery KJ (2004) Path integration in mammals. Hippocampus 14:180–192.

Etienne AS, Boulens V, Maurer R, Rowe T, Siegrist C (2000) A brief view of known landmarks reorientates path integration in hamsters. Naturwissenschaften 87:494–498.

Foldiak P (2002) Sparse coding in the primate cortex. In: The handbook of brain theory and neural networks, Ed 2 (Arbib M, ed), pp 1064–1068. Cambridge, MA: MIT.

Foster DJ, Morris RG, Dayan P (2000) A model of hippocampally dependent navigation, using the temporal difference learning rule. Hippocampus 10:1–16.

Franzius M, Vollgraf R, Wiskott L (2007) From grids to places. J Comput Neurosci 22:297–299.

Fuhs MC, Touretzky DS (2006) A spin glass model of path integration in rat medial entorhinal cortex. J Neurosci 26:4266–4276.

Fyhn M, Molden S, Witter MP, Moser EI, Moser MB (2004) Spatial representation in the entorhinal cortex. Science 305:1258–1264.

Fyhn M, Hafting T, Treves A, Moser MB, Moser EI (2007) Hippocampal remapping and grid realignment in entorhinal cortex. Nature 446:190–194.

Gorchetchnikov A, Grossberg S (2007) Space, time and learning in the hippocampus: how fine spatial and temporal scales are expanded into population codes for behavioral control. Neural Netw 20:182–193.

Hafting T, Fyhn M, Molden S, Moser MB, Moser EI (2005) Microstructure of a spatial map in the entorhinal cortex. Nature 436:801–806.

Hahnloser RH, Kozhevnikov AA, Fee MS (2002) An ultra-sparse code underlies the generation of neural sequences in a songbird. Nature 419:65–70.

Hartley T, Burgess N, Lever C, Cacucci F, O'Keefe J (2000) Modeling place fields in terms of the cortical inputs to the hippocampus. Hippocampus 10:369–379.

Hollup SA, Molden S, Donnett JG, Moser MB, Moser EI (2001) Accumulation of hippocampal place fields at the goal location in an annular watermaze task. J Neurosci 21:1635–1644.

Kolton L, Camhi JM (1995) Cartesian representation of stimulus direction: parallel processing by two sets of giant interneurons in the cockroach. J Comp Physiol [A] 176:691–702.

Kuipers B (2000) The spatial semantic hierarchy. Artif Intell 119:191–233.

Laurent G, Stopfer M, Friedrich RW, Rabinovich MI, Volkovskii A, Abarbanel HD (2001) Odor encoding as an active, dynamical process: experiments, computation, and theory. Annu Rev Neurosci 24:263–297.

Lopez-Barneo J, Darlot C, Berthoz A, Baker R (1982) Neuronal activity in prepositus nucleus correlated with eye movement in the alert cat. J Neurophysiol 47:329–352.

McNaughton BL, Barnes CA, O'Keefe J (1983) The contributions of position, direction, and velocity to single unit activity in the hippocampus of freely-moving rats. Exp Brain Res 52:41–49.

McNaughton BL, Barnes CA, Gerrard JL, Gothard K, Jung MW, Knierim JJ, Kudrimoti H, Qin Y, Skaggs WE, Suster M, Weaver KL (1996) Deciphering the hippocampal polyglot: the hippocampus as a path integration system. J Exp Biol 199:173–185.

McNaughton BL, Battaglia FP, Jensen O, Moser EI, Moser MB (2006) Path integration and the neural basis of the "cognitive map." Nat Rev Neurosci 7:663–678.

Miller P, Clesceri N (2002) Waste sites as biological reactors: characterization and modeling. Boca Raton, FL: CRC.

Mittelstaedt M, Mittelstaedt H (1980) Homing by path integration in a mammal. Naturwissenschaften 67:566–567.

Mohan PA (2002) Residue number systems: algorithms and architectures. Boston: Kluwer Academic.

Mulders WH, West MJ, Slomianka L (1997) Neuron numbers in the presubiculum, parasubiculum, and entorhinal area of the rat. J Comp Neurol 385:83–94.

Muller RU, Kubie JL, Ranck BJ Jr (1987) Spatial firing patterns of hippocampal complex-spike cells in a fixed environment. J Neurosci 7:1935–1950.

O'Keefe J, Burgess N (1996) Geometric determinants of the place fields of hippocampal neurons. Nature 381:425–428.

O'Keefe J, Burgess N (2005) Dual phase and rate coding in hippocampal place cells: theoretical significance and relationship to entorhinal grid cells. Hippocampus 15:853–866.

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.

O'Keefe J, Nadel L (1978) The hippocampus as a cognitive map. Oxford: Oxford UP.

Parron C, Save E (2004) Evidence for entorhinal and parietal cortices involvement in path integration in the rat. Exp Brain Res 159:349–359.

Perez-Orive J, Mazor O, Turner GC, Cassenaer S, Wilson RI, Laurent G (2002) Oscillations and sparsening of odor representations in the mushroom body. Science 297:359–365.

Recht M (1988) The biology of domestic rats: telemetry yields insights for pest control. Paper presented at the Thirteenth Vertebrate Pest Conference. University of Nebraska, Lincoln, NE, March.

Redish AD (2001) The hippocampal debate: are we asking the right questions? Behav Brain Res 127:81–98.

Redish AD, Touretzky DS (1997) Cognitive maps beyond the hippocampus. Hippocampus 7:15–35.

Redish AD, Touretzky DS (1998) The role of the hippocampus in solving the morris water maze. Neural Comput 10:73–111.

Rolls ET, Stringer SM, Elliot T (2006) Entorhinal cortex grid cells can map to hippocampal place cells by competitive learning. Network 17:447–465.

Russell JC, Towns DR, Anderson SH, Clout MN (2005) Intercepting the first rat ashore. Nature 437:1107.

Samsonovich A, McNaughton BL (1997) Path integration and cognitive mapping in a continuous attractor neural network model. J Neurosci 17:5900–5920.

Soderstrand M, Jenkins W, Jullien G, Taylor EFJ (1986) Residue number system arithmetic: modern applications in digital signal processing. New York: IEEE.

Solstad T, Moser EI, Einevoll GT (2006) From grid cells to place cells: a mathematical model. Hippocampus 16:1026–1031.

Sun H, Yao TR (1994) A neural-like network approach to residue-to-decimal conversion. Neural Netw 6:3883–3887.

Taube JS, Muller RU, Ranck BJ Jr (1990) Head-direction cells recorded from the postsubiculum in freely moving rats. I. description and quantitative analysis. J Neurosci 10:420–435.

Touretzky DS, Redish AD (1996) Theory of rodent navigation based on interacting representations of space. Hippocampus 6:247–270.

Tsodyks M (1999) Attractor neural network models of spatial maps in hippocampus. Hippocampus 9:481–489.

Whishaw IQ, Maaswinkel H (1998) Rats with fimbria-fornix lesions are impaired in path integration: a role for the hippocampus in "sense of direction." J Neurosci 18:3050–3058.

Wilson MA, McNaughton BL (1993) Dynamics of the hippocampal ensemble code for space. Science 261:1055–1058.

Witter MP, Groenewegen HJ (1984) Laminar origin and septotemporal distribution of entorhinal and perirhinal projections to the hippocampus in the cat. J Comp Neurol 224:371–385.