Grid cells generate an analog error-correcting code for singularly precise neural computation

Sameet Sreenivasan[1,2] & Ila Fiete[1]

Entorhinal grid cells in mammals fire as a function of animal location, with spatially periodic response patterns. This nonlocal periodic representation of location, a local variable, is unlike other neural codes. There is no theoretical explanation for why such a code should exist. We examined how accurately the grid code with noisy neurons allows an ideal observer to estimate location and found this code to be a previously unknown type of population code with unprecedented robustness to noise. In particular, the representational accuracy attained by grid cells over the coding range was in a qualitatively different class from what is possible with observed sensory and motor population codes. We found that a simple neural network can effectively correct the grid code. To the best of our knowledge, these results are the first demonstration that the brain contains, and may exploit, powerful error-correcting codes for analog variables.

The cortex is a noisy system. Spikes emitted in response to repeated stimuli are often stochastic, with a Fano factor close to 1 (ref. 1), and synaptic activation is probabilistic[2]. Noise may be advantageous in certain learning contexts[3–5]. However, if the aim is to perform accurate representation and computation, noise can be a serious impediment, an especially severe one if the computation to be performed is recursive (**Supplementary Fig. 1**). In such cases, noise in every time step can accrue and propagate. If the variable being computed is bounded, the noise will swamp the computation, rendering it useless. Integration is one common example of a recursive computation; the state of the integrated variable must be held faithfully in time and incremented in response to inputs. If the duration of integration is long, accumulating errors can easily exceed the integrated size of the variable.

Population codes, many neurons collectively encoding a given variable, permit a more faithful representation than a single neuron can. However, the population codes commonly found in sensory, motor and some cognitive areas[6–10] enable only weak improvement in coding an analog variable as a function of neuron number (N): the squared error declines at best as $1/N$ (or as $1/N^2$, if the tuning curve widths shrink as $1/N$, which is optimal)[11–16]. We refer to all population codes with such a polynomial scaling of inverse squared error with N as classical population codes (CPCs). CPCs are a broad category: populations in which the tuning curves for all neurons are smooth and identical up to translation, including unimodal tuning curves, monotonic tuning curves or even periodic tuning curves with a single period[17], are all CPCs. CPCs also include some populations with unimodal tuning curves that differ by both translations and scalings (see **Supplementary Results**, section 4).

Furthermore, the weak accuracy gains with neuron number in CPCs are obtained through the inefficient use of neural redundancy. To quantify the efficiency of a redundant code, we define the information rate $\rho \in [0,1]$ as the ratio of the number of information bits divided by the total number of conveyed bits[18]. In CPCs, the mutual information between the code and signal scales as $\sim \log N$. (In such codes, the mutual information for large N scales as the log of the Fisher information J (ref. 15), which itself scales as $\sim N$ (ref. 11)). However, the total number of conveyed bits per time step scales as $\sim N$. Thus, the information rate ρ of CPCs scales as $\sim \log N / N$, which approaches zero for large N. In summary, CPCs can attain asymptotically zero error, but at the cost of an asymptotically zero information rate.

In this sense, CPCs are similar to the naive repetition code (for discrete variables). Consider the problem of representing a single bit, if each encoding bit has a probability $p < 0.5$ of flipping. If the same bit is encoded N times, for example, and decoding is by majority vote, the repetition code produces errors with a probability of $\sim p^N$. But $\rho = 1/N$ tends to zero asymptotically with N. For both the CPC and repetition code, declining error (magnitude or probability, respectively) comes at the price of asymptotically zero information rate.

Is it possible to do qualitatively better? Until 1948, this was believed to be impossible. Then Shannon's Fundamental Theorem revealed that, astonishingly, there exist codes that, for discrete variables, allow asymptotically zero error probability at asymptotically nonzero information rates[19]. For analog variables, a 'good' code in this Shannon sense can produce exponentially small error at asymptotically finite information rates[18,20].

But does the brain contain good codes (**Fig. 1a**)? We analyzed the peculiar encoding of location in grid cells and found that estimation error divided by the coding range goes to zero at an asymptotically finite information rate, unlike in CPCs. Moreover, the error divided by range decreases exponentially with neuron number, rather than polynomially, as is seen with CPCs. The leap in performance of the grid code over CPCs parallels, for analog variables, the leap in performance from repetition-type codes to modern codes following Shannon's Fundamental Theorem. Thus, the grid code helps to define a class of neural population codes that we refer to as exponentially strong population codes (EPCs).

[1]Center for Learning and Memory, University of Texas at Austin, Austin, Texas, USA. [2]Department of Computer Science, Rensselaer Polytechnic Institute, Troy, New York, USA. Correspondence should be addressed to I.F. (ilafiete@mail.clm.utexas.edu).

RESULTS

The observed grid code for animal location: formalization

Grid cells are thought to be the neural substrates for path integration, or estimating changing location on the basis of self-motion cues[21–24]. Each cell fires as a function of the animal's location in two-dimensional space, with a firing peak arranged on every vertex of a regular triangular lattice that tiles the explored space (one-dimensional schematic in **Fig. 1b**). Cells with a common spatial period also share a common grid orientation; their responses differ only by spatial translations, or different preferred firing phases, with respect to their common response period (**Fig. 1b**). We refer to these cells as belonging to one grid network. The entire activity pattern in one grid network specifies (two-dimensional) location only as a (two-dimensional) spatial phase. The network response cannot distinguish locations separated by integer multiples of the spatial period along the primary grid axes[25] (**Fig. 1b**). We limit what follows to one dimension; animal location x varies along a line in space and each cell's tuning curve is a periodic one-dimensional spatial pattern (the results generalize to two dimensions; **Supplementary Results**, section 13).

If network activity were a perfect reflection of location x, rendered as a spatial phase with respect to period λ_α, it would provide the following information about location[25]:

$$\underline{\phi}_\alpha(x) = \frac{x \bmod \lambda_\alpha}{\lambda_\alpha} \equiv \frac{x}{\lambda_\alpha} \bmod 1 \qquad (1)$$

where $\underline{\phi}_\alpha \in [0,1)$ is the true phase associated with location x and period λ_α. In actuality, there is error in what the grid network conveys about x. The readout error ($\xi_\alpha^R(t)$, a zero-mean Gaussian

noise) reflects the uncertainty inherent in deducing network phase from a finite number of neural spikes[11–16,26]. The integration error ($\xi_\alpha^I(t) = \sum_{t'=0}^{t} \zeta_\alpha^I(t')$, where each $\zeta_\alpha^I(t')$ is zero-mean Gaussian, t is the instantaneous time, and $t = 0$ marks the beginning of velocity integration starting at a known location) reflects the mismatch between true animal location $x(t)$ and the phase-encoded location estimate in the grid network. The mismatch arises if the network's estimate derives from noisy neural integration of animal velocity. As in any neural integrator[24,27–29], the integration error is cumulative over time even if the velocity signal is error-free. Thus, the spatial data in each grid network is

$$\phi_\alpha(x,t) = \left(\frac{x(t)}{\lambda_\alpha} + \xi_\alpha^I(t) + \xi_\alpha^R(t) \right) \bmod 1 \qquad (2)$$

In what follows, the theoretical analysis is, for convenience, based on this network phase, whereas numerical results are based on the multi-peaked spatial firing rate vectors of all cells (Online Methods). Both quantities convey the same information about x.

The entorhinal cortex contains multiple grid networks, with different response periods[21,30]. Assuming N networks with periods $\lambda_1, \ldots, \lambda_N$, the grid population code (GPC) is defined as the vector of N spatial phases[25]

$$\vec{\phi}(x,t) = (\phi_1(x,t),\ldots,\phi_N(x,t)) \qquad (3)$$

Although the different periods are distinct, they span only a decade in scale (ranging from ~0.3–3 m)[21,30] and are very small compared with the typical distances (100–1,000 m) a rat covers while foraging over a day. In this sense, the periods share roughly the same size magnitude, which we refer to as λ.

Figure 1 Coding for location: GPC and CPCs. (**a**) The general problem of (noisy channel) coding. A variable x is represented in some way, as $\vec{\phi}(x)$, but the representation is subject to noise with a given distribution. The problem is to find efficient representations that allow an ideal observer to most accurately estimate x. (**b**) Top, schematic spatial tuning curves of three grid cells (light gray, gray and black) with identical response periods (λ_α) and different preferred phases (ϕ_i^*). Bottom, current animal location. Right, the instantaneous error-free firing rates of all grid cells in one network, arranged by preferred phase, encode current animal location as an instantaneous network phase; the error-free instantaneous network phase is $\underline{\phi}_\alpha(x(t)) = 3/4$. (**c**) Top, N independent M-neuron networks, encoding N instantaneous phases. Each phase has ~M coding states. Together, the networks provide ~M^N coding states. Bottom, one CPC network of NM neurons, with unimodal tuning curves, encodes location as a single phase with a resolution of ~NM states. (**d**) Firing rates of three random cells, from the CPC network (bottom) or from three different grid networks (top), as x is varied. (**e**) Sample of (ideally) decoded locations obtained from the slightly perturbed phases $\vec{\phi} = \vec{\phi}(x_0) + \vec{\xi}$ for the GPC (black) and CPC (green), for true location x_0. Noise $\vec{\xi}$ (inset) is Gaussian. $R_{CPC} = R = 90,090$ cm, $N = 5$, $\sigma_\alpha = 0.04$, $\vec{\lambda} = \{10, 14, 18, 22, 26\}$ cm. Parameters are common to **e** and **f**. (**f**) Data are generated as in **c**, but encoding and decoding are restricted to an interval of size $R_I \ll R$ around x_0. $R_{CPC} = R_I = 500$ cm.
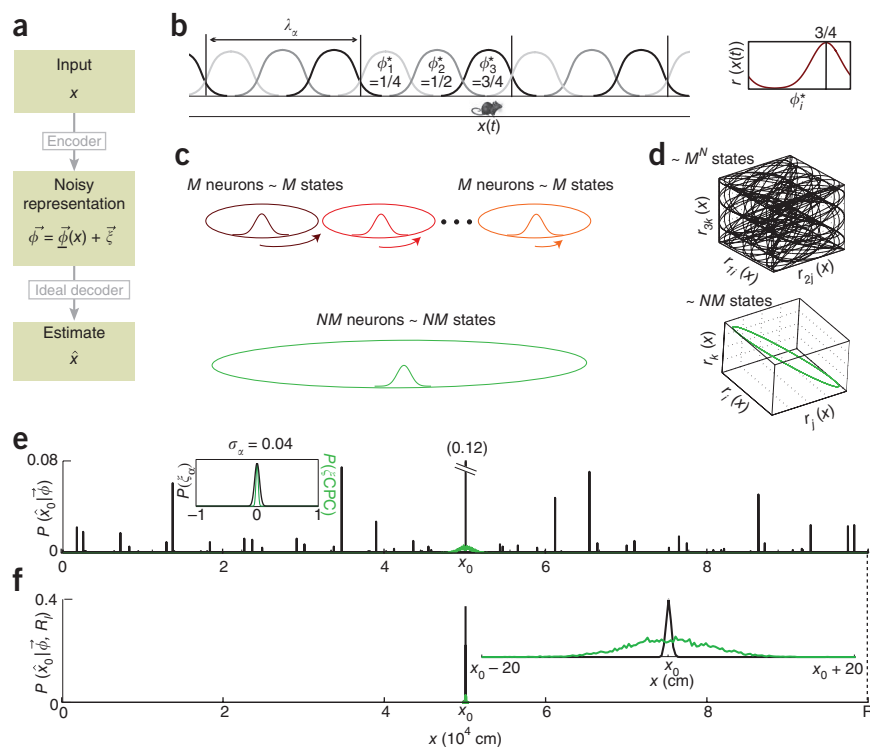
Table 1 Comparable setups for encoding by the GPC and CPC

	GPC	CPC
Number of networks	N	1
Neurons per network	M	NM
Total neuron number	NM	NM
Coding range	R or R_l	$R_{CPC} = R$ or R_l
Spatial response period per network	λ_α $(<<R_l, R)$	R_{CPC}
Noiseless (true) phase per network	$\underline{\phi}_\alpha = (x/\lambda_\alpha) \bmod 1$	$\underline{\phi}_{CPC} = (x/R_{CPC}) \bmod 1$
Number of response peaks over range $R_l, R_{CPC} = R_l$	R_l/λ_α in α^{th} network	1
Total phase noise variance per network	σ_α^2	$\sigma_{CPC}^2 = \sigma_\alpha^2 / N$

The GPC for animal locations in a large foraging range, as defined by the phases of equations (2) and (3), is analogous to encoding the time of day with a set of several short-period clocks (for example, with 2–20 min periods), instead of with a single clock. For comparison, a place cell–like CPC with unimodal tuning curves over the foraging range is similar to one 24-h clock (Online Methods and **Table 1**). Frequently, over the course of the day, the hand of every clock is jittered around its present value, with standard deviation a fixed fraction of that clock's period. Is there any possible advantage to representing time with multiple clocks?

Range grows exponentially with neuron number

A CPC, and, in particular, each grid network individually, can uniquely represent linearly larger ranges at the same absolute resolution, with linearly more neurons (**Fig. 1c**). However, the combined GPC with all phases can uniquely represent exponentially larger ranges with linearly more neurons at constant absolute resolution (fixed tuning curve width and density of tuning curve centers)[25]. Intuitively, as x is varied, the phases in different networks vary at different rates ($\sim x/\lambda_\alpha$; **Fig. 1c**). If the ratios of the periods are not small rational numbers, the total phase vector of equation (3) will run through a vast combination of phase components over a large range of x before getting close to overlapping a previously visited state, resulting in dense coverage of the volume of the space of possible neural firing rates (**Fig. 1d**). In contrast, a CPC defines a single phase that traces a very sparse loop through the space of neural rates as x is varied. This space-filling intuition for the GPC versus the CPC is the same whether we plot the neural firing rates (**Fig. 1d**) or the more abstract location phases (data not shown).

Formally, the range R of locations possessing a unique GPC representation at fixed absolute resolution is exponentially large[25]:

$$R \sim \lambda \Delta\phi \left(\frac{1}{\Delta\phi}\right)^{\gamma N} = \lambda e^{\beta N} >> \lambda \qquad (4)$$

$\Delta\phi$ is the finest discriminable phase difference in each network (this is the inverse resolution), and $\gamma \lesssim 1$ depends on the specific choice and spacing of the periods λ_α[25]. N is the number of networks and is linearly proportional to the total neuron number (**Table 1**). For $N >> 1$ and $1/\Delta\phi > e$, it follows that $\beta > \gamma$. For conservative choices of N and $\Delta\phi$, the range of locations represented by the GPC exceeds 10^5 km[25].

Extreme noise sensitivity

However, this high-range GPC is pathologically sensitive to noise. Consider $\vec{\phi}$ (equation (2)) instead of $\vec{\underline{\phi}}$ (equation (1)), with a total phase noise variance of σ_α^2 per GPC network (and σ_α^2/N in the CPC; see Online Methods and **Table 1**). Very small phase noise in the GPC results in massive, nonlocal errors in (ideally) decoded location

(**Fig. 1e**). In this typical sample of the posterior location distribution given the noisy phases, errors are vast, of size $\sim R$. In contrast, small noise in CPC phase produces local and relatively small location errors (**Fig. 1e**). When comparing the GPC and the CPC, we always used the same coding range for location x, encoded using the same total neuron number and the same single-neuron noise (Online Methods and **Table 1**). The CPC allows superior position estimation compared with the massive errors obtained using the GPC (**Fig. 1e**). Thus, the GPC's noise sensitivity appears to be an intolerable fault. In what follows, we show that the GPC's exponentially large representable range and, counterintuitively, extreme noise sensitivity enables extraordinarily precise location estimation.

Extraordinary noise robustness over reduced coding range

Suppose that, over its lifetime, an animal only covers a subrange $[0, R_l]$ of the representable range. R_l is assumed to satisfy $R_l << R$ and $R_l >> \lambda$, and can be very large given the vastness of R. Let the ideal decoder of the noisy GPC be similarly constrained; given the phase, it picks the most likely location interpretation from $[0, R_l]$ instead of from $[0, R)$ as described above. The absolute error in both the GPC and CPC improved when the coding range was reduced, as we observed in the posterior sampling of estimated locations (**Fig. 1f**). The CPC peak shrank in width by the large factor $(R_l/R)^2$.

However, the GPC then grossly outperformed the CPC. The noisy GPC phases that formerly mapped to scattered peaks (**Fig. 1f**) mapped onto a single peak when decoded. This peak was centered at the true location and was much narrower than the CPC peak (**Fig. 1f**).

## Abstract structure of the GPC: how it suppresses noise

Noise sensitivity over the representable range and robustness over reduced ranges result from two complementary features of the mapping between location and GPC phase. Suppose that locations are discrete (for example, separated by $\Delta x = \lambda/6$). Extreme noise sensitivity (**Fig. 1c**) results from the fact that neighboring points in the space of GPC phases correspond to very separated locations (**Fig. 2a**); if a phase is perturbed to a neighboring point in phase space by noise, the distance between true location and the location represented by the perturbed phase is large.

A complementary feature of the GPC location-phase mapping is that nearby locations map to very separated phases (**Fig. 2b**). For example, translating location x by the modest displacement $\lambda/2 << R, R_l$ corresponds to a phase change of $(\lambda/2\lambda_1, \lambda/2\lambda_2, ..., \lambda/2\lambda_N)$. Because each $\lambda_\alpha \approx \lambda$ in size, this is a change of $\sim 1/2$ in each phase component. The slightly translated location thus maps to a point maximally far in phase space from the point to which x mapped. Indeed, all locations in any subinterval of length $R_l << R$, as a result of their relative proximity to each other, map to mutually well-separated points in phase space (**Fig. 2c**). We refer to R_l as the legitimate range for location and the corresponding noise-free GPC phases as the codewords.

This dual near-to-far mapping from locations to phases and phases to locations creates the code's interleaving structure: each codeword is surrounded by, and separated from, all of the other codewords by a cushion of phase space corresponding to illegitimate locations (**Fig. 2c**). For any $R_l << R$, the separation between codewords is finite, with a minimum pair-wise distance of $d_{min}(R, R_l) > 0$ between codewords (**Fig. 2c**). It follows that any noise of amplitude smaller than $d_{min}/2$ is exactly correctable simply by mapping the perturbed representation to the nearest codeword. Thus, finite minimum distance is an important characteristic of certain strong error-correcting codes.

This code's information rate is $\rho = \log(R_l/\Delta x)/\log(R/\Delta x)$. Equivalently, any choice of information rate dictates the legitimate

Figure 2 Error correction and the structure of the GPC location-phase map. Left, real space. Right, phase space. Phase space is an *N*-dimensional cube of with sides of length 1 and periodic boundary conditions. (**a**) Near-to-far mapping from phase to locations. A location (blue) maps to a point (blue) in phase space, but neighboring phases (gray) map to locations very distant from the original location. (**b**) Near-to-far mapping from locations to phase. Two nearby locations (light blue, blue) separated by about $\lambda/2$ map to distant phases (halfway across the cube is maximally distant because of the cube's periodic boundaries). (**c**) Minimum distance property. Discrete locations in a small interval $R_l << R$ all map to well-separated points in phase (codewords), with minimum distance d_{min} between them. (**d**) Continuous locations *x* are represented by a continuous coding line wrapping around phase space. With phase space rendered as an *N*-dimensional cube with periodic boundaries, the line looks like separate segments. The segments for any location in $R_l << R$ are well-spaced, with a finite minimum perpendicular distance d_{min}. Inset, true phase (white dot), the perturbing noise vector (red) and the corrected phase (tip of black arrow) are shown. After correction, the residual phase error (black double-headed arrow) is along the true coding line. GPC location error is approximately this residual error times λ. (**e**) The CPC. The phase error was smaller by a factor of $1/N$ than the length of the GPC perturbing vector. Total CPC location error is this phase error times R_l.
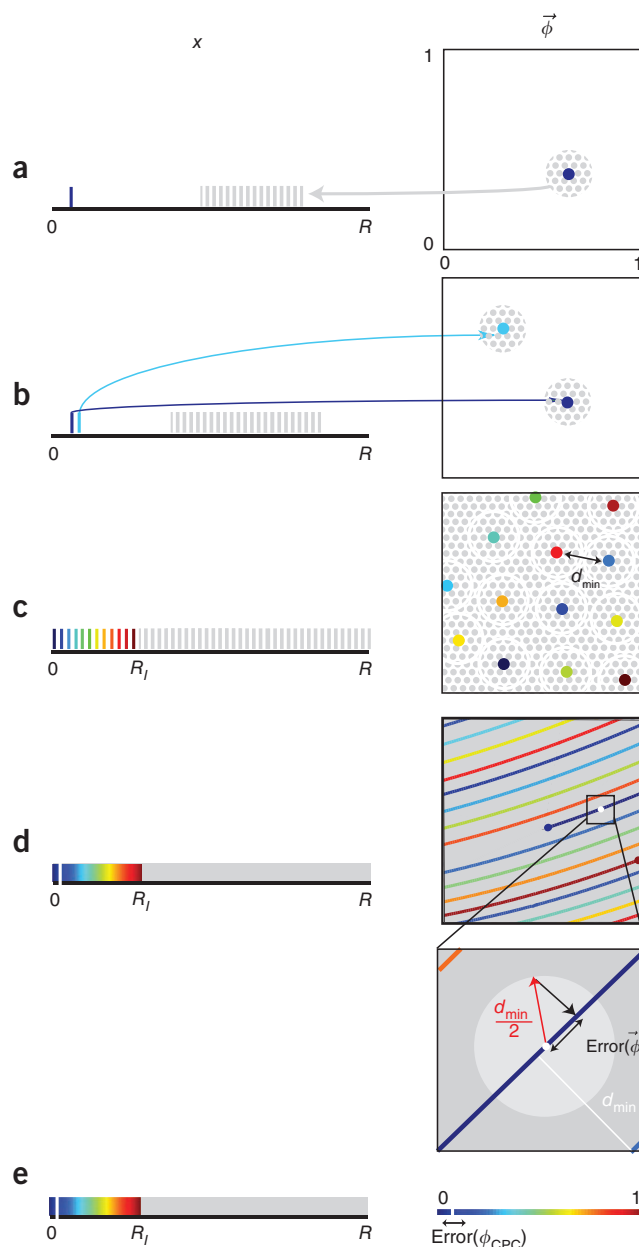
location range, $(R_l / \Delta x) = (R / \Delta x)^\rho$. The minimum distance, and thus the maximal correctible error, both increase with decreasing information rate. This discrete GPC location code operating at information rate < 1 enables exact correction of any noise smaller than the corresponding minimum distance.

However, location and the spatial phases represented by the GPC are continuous variables. The basic principles underlying error control remain similar to those from the discrete case (see below), and, although error correction cannot be exact for analog variables, it is exponentially strong.

As *x* increases continuously from a starting value, the GPC phase vector (equation (1)) traces a continuous line that wraps around phase space. A component of the phase vector wraps once, whenever *x* has moved by the corresponding network period, causing the phase vector to exit one side of the phase cube and re-enter at the opposite side. Thus, the continuous phase vector line rendered in the phase cube looks like a set of parallel line segments (**Fig. 2d**), each representing an interval of locations of length $\sim\lambda$. After *x* has traversed a distance *R*, the phase space volume is fully covered, once over, by the phase coding line, assuming the coding line has thickness $\Delta\phi$ (equation (4); this defines R)[25].

As in the discrete case, the continuous GPC possesses an interleaving phase-space structure. Initially, as *x* moves from a starting value, the coding phase vector warps, re-emerging maximally far from existing segments. All coding segments for any small subinterval of length *L*, $\lambda << L << R$ are very well-separated and coarsely, but evenly, encircle phase space. As *x* increases further, the additional phase line segments fall in between the existing ones. As *x* increases further still, the new phase line segments again fall in between the preceding sets, interleaved with them, and so on. As a result, neighboring phase segments correspond to remote locations (noise sensitivity). Relatively nearby locations (separated by $>\lambda$) are represented by phase line segments that are disperse. For any legitimate range $R_l << R$, the codewords are now the corresponding continuous phase line segments. Again, there is a notion of minimum distance $d_{min}(R, R_l) > 0$: the smallest perpendicular distance between all pairs of coding line segments (**Fig. 2d**). This code's information rate is $\rho = \log(R_l / \lambda\Delta\phi)/\log(R / \lambda\Delta\phi)$.

If any codeword is perturbed by noise $< d_{min}/2$, then assigning the perturbed phase to its nearest codeword is guaranteed to place it somewhere on the true coding line segment (**Fig. 2d**). This corrected phase may not be exactly the true phase (**Fig. 2d**), but, given that each phase segment represents a distance of $\sim\lambda$ (because of phase wrapping),

the absolute error in location estimation is smaller than $\sim\lambda$. This remains true for any R_l (for phase noise $< d_{min}(R, R_l)/2$). Thus, for any $R_l >> \lambda$, it follows that the fractional error, that is, the ratio of the absolute error divided by the coding range R_l, is small and becomes smaller as R_l becomes larger.

More precisely, we computed the fractional error as follows. A random phase noise vector of length *a* (assume $a < d_{min}/2$), projected onto one dimension (here, onto the correct one-dimensional coding phase segment), typically has length $a/\sqrt{N}$ (**Fig. 2d**). Any two phases on a single phase segment, separated by *b* in phase, represent real-space locations separated by $\sim\lambda b$ (on a single segment, phase distance is real-space distance divided by period). Thus, the fractional error in location estimation from the largest correctible noise is

$$\frac{\text{error}_{GPC}(x)}{R_l} \sim \frac{a}{\sqrt{N}}\frac{\lambda}{R_l} \sim \frac{a}{\sqrt{N}}e^{-\rho\beta N} \qquad (5)$$

where $\text{error}_{GPC}(x)$ is the root mean square error in estimating location. To obtain the last scaling, we combined equation (4) and the
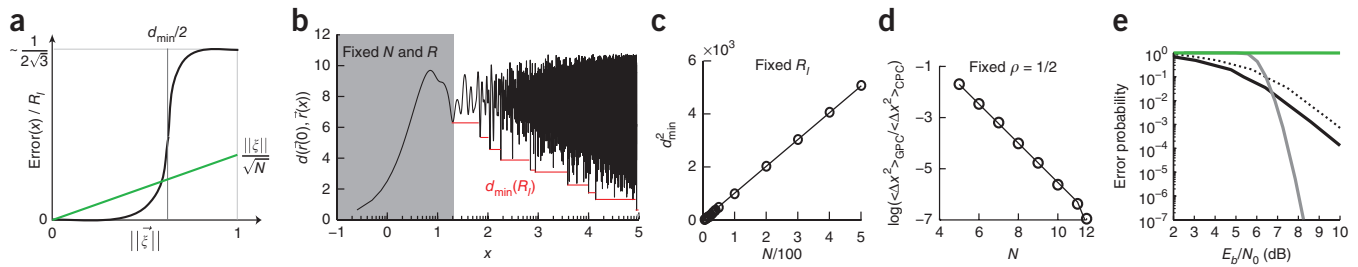
Figure 3 Scaling of d_{min} and error correction. (**a**) Error versus noise amplitude (schematic, finite N). GPC (black) enables excellent error control for noise below a threshold ($d_{min}/2$) and provides almost no information above that threshold. CPC performance (green) degrades linearly with noise. (**b**) The distance between vectors $\vec{r}(0)$ and $\vec{r}(x)$, as x increases from 0 to R (x in cm, plotted on a $\log_{10}$ scale). At a short displacement, $x = \lambda/2$, $\vec{r}(x)$ is far from $\vec{r}(0)$; for larger displacements, it is closer, then further, and so on (interleaving property). As x grows, the approaches to $\vec{r}(0)$ get closer still, giving the distance curve a monotonically decreasing lower envelope (red). The red line represents the minimum distance d_{min} of the code if $R_l = x$. As $R_l = x$ increases toward R, the minimum distance of the code decreases toward 0. (**c**) The squared minimum distance d_{min}^2 grows with N, if R_l is held fixed. Circles indicate numerical results and the black line is the linear fit, consistent with linear prediction by theory (**Supplementary Results**, section 7). (**d**) Mean squared location error ratio for the GPC and CPC (obtained by sampling noisy phases and maximum likelihood decoding over R_l) decreased exponentially with N (circles), consistent with calculations. The gray line is the fit, qualitatively predicted by equation (7). (**e**) Probability of error in decoded location as a function of the signal-to-noise ratio (SNR) per information bit (E_b/N_0) for the GPC (solid black line, for $\lambda\Delta\phi \approx 0.1$ cm, resulting in $\rho \approx 0.56$; dotted black line, for $\lambda\Delta\phi \approx 1$ cm, resulting in $\rho \approx 0.5$), CPC (green), and a rate $\rho = 0.935$ Reed-Solomon code (gray). At low E_b/N_0, GPC outperforms the Reed-Solomon code. (CPC error declines weakly with increasing E_b/N_0.)

information rate definition. For any coding range choice $R_l \gg \lambda$, the fractional error of the grid code is very small, suppressed by the factor λ/R_l. The bigger the R_l, or equivalently, the smaller the scale λ of the grid periods relative to R_l, the smaller the fractional error of the GPC. At a fixed information rate, the fractional error vanishes exponentially strongly (equivalently, the inverse fractional error or dynamic range of the code, grows exponentially) as N is increased.

We then fairly compared a phase noise vector of length a in the GPC with a smaller noise of amplitude a/N in the CPC phase (Online Methods and **Table 1**). However, this phase error in the CPC produces a large absolute location error (root mean square error), given by $\text{error}_{CPC}(x) \sim R_l a / N$ (the scaling with R_l arises because CPC phase is simply x divided by the coding range; **Fig. 2e**). Compare an absolute location error of $\sim \lambda a / \sqrt{N}$ for the GPC (above). The fractional location error of the CPC is

$$\frac{\text{error}_{CPC}(x)}{R_l} \sim \frac{a}{N} \qquad (6)$$

The ratio of (both fractional and absolute) GPC to CPC location errors over a range R_l with identical encoding constraints is given by equations (5) and (6):

$$\frac{\text{error}_{GPC}(x)}{\text{error}_{CPC}(x)} \sim \sqrt{N}\frac{\lambda}{R_l} \sim \sqrt{N}e^{-\rho\beta N} \qquad (7)$$

In other words, over any fixed coding range $R_l \gg \lambda$, the GPC greatly outperforms the CPC (first scaling relationship of equation (7)). Furthermore, over equal coding ranges, the GPC outperforms the CPC exponentially strongly with N as N is increased at a fixed GPC information rate (second scaling relationship of equation (7)). The square of the error ratio of equation (7) gives the ratio of rates at which the squared integration error grows during a trajectory for the GPC versus the CPC.

In the clocks analogy, multiple short-period noisy clocks, properly decoded, can be used to tell time over a day with far greater precision than a corresponding 24-h noisy clock.

Maximal correctible error and numerical validation
Characteristic of strong error-correcting codes, and in contrast with CPCs, errors in the GPC below a threshold ($d_{min}/2$) can be strongly suppressed, but above the threshold, practically no information can be recovered (**Fig. 3a**). What sets this threshold? If $R_l = R$, then the

information rate is 1 (no redundancy) and no error correction is possible, suggesting that $d_{min}(R, R_l) = 0$. As $\rho \to 0$ (for example, $R_l \to 0$ for fixed R, or $R \to \infty$ for fixed R_l), coding redundancy increases and the threshold increases (larger errors are correctible) (**Fig. 3b**). We then derived analytically (**Supplementary Results**, section 7) and verified numerically (**Fig. 3c**) that, if R_l is held fixed as N (and thus R) increases, then d_{min} increases as $\sqrt{N}$. Thus, for any given noise amplitude and fixed R_l, it is possible to essentially avoid coding failure with sufficiently large N (enough distinct grid periods).

Alternatively, if ρ is held fixed as N increases, then d_{min} remains constant while R and R_l increase exponentially with N (data not shown). The same noise can then be corrected to obtain the same absolute location error over exponentially larger coding ranges R_l. Thus, the fractional error of the GPC decreases exponentially with N for fixed maximal noise amplitude.

Finally, we verified numerically that, with phase noise $< d_{min}/2$, the ratio of location error from the GPC and CPC over equal coding ranges decreased exponentially with N (**Fig. 3d**), as predicted in equation (7).

Fisher information
Fisher information, a local measure of inverse least mean squared error, is commonly used to study population codes[11–16,31]. Thus, it is useful to also understand using Fisher information why the GPC outperforms the CPC (derivation in **Supplementary Results**, section 4), even if, in contrast with the geometric derivation already given, it cannot explain global features of the code, including the identifiability of coded location over exponentially large ranges, or when the code breaks down, or why. Fisher information grows with the steepness of neural tuning curves. In a unimodal CPC with fixed neuron number, inverse Fisher information increases as the coding range squared, as the tuning curves must become broader to cover the range. In each grid network, however, the number of response peaks grows with the coding range, but not their widths; thus, inverse Fisher information and absolute least squared error are unchanged. The Fisher information result also shows explicitly that the GPC advantage is not attributable to sparseness; the GPC outperforms the CPC by similar margins when both codes have the same lifetime sparseness (the fraction of the coding range over which each neuron is active). Varying the relative sparseness modulates the error ratio of the two codes (equation (7))
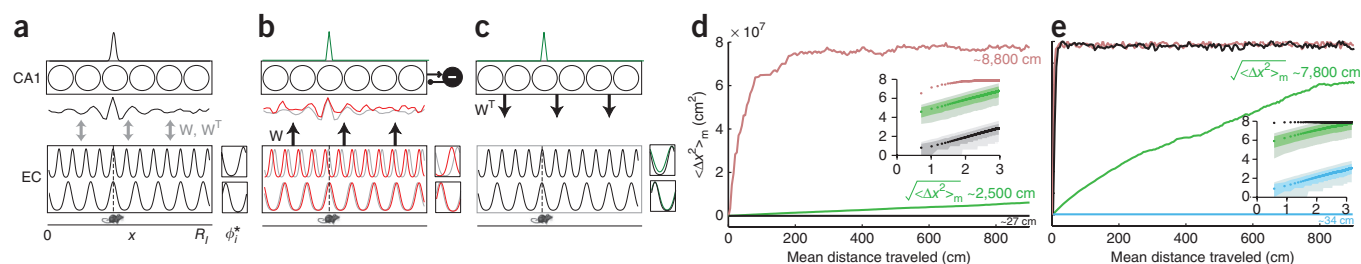
Figure 4 Simple neural network can perform ongoing error correction for accurate path integration. (**a**–**c**) Network and activity schematic. Bottom, animal location. Middle, spatial tuning curve of maximally activated grid cells for present rat location, from two entorhinal cortex (EC) networks. Right, instantaneous activation of all grid cells in the network, as a function of preferred spatial phase ϕ_i^*. Arrows indicate grid cell–readout weights (W) and readout–grid cell weights (W^T). Depicted above the arrows are the summed inputs to the readout (CA1) cells. Top, output activity of readout neurons. During learning, the correct grid cells were activated (**a**). These cells wire up to the active readout cell. After learning, erroneous grid cell activation (red) drives readout cells (red summed input) (**b**); error-free curves are shown in gray for comparison. Winner-take-all dynamics selects a winner readout cell (top), which corrects activity in the grid cell network (**c**). (**d**,**e**) Median squared error growth during path integration over a simulated trajectory (100 trials). Without correction (red; decoding over $R_l \ll R$, but no correction of grid phases after decoding), GPC location estimation is useless (error > displacement), even though individual phase errors grow slowly. Green, CPC; black, grid cell–readout–grid cell closed-loop continuous operation. Inset, log-log plot; envelopes indicate where 25–75% and 10–90% of all 100 trials lie, respectively. The ratio of slopes (diffusion constants) of CPC and closed-loop trajectories was ≈ 10^4 in **d**. Data in **e** are generated as in **d**, but with fivefold larger amplitude phase noise. Phase noise in one error-correcting iteration frequently exceeded $d_{\min}/2$, so error correction failed (black). Blue, readout with continuity constraint. The ratio of slopes for the CPC and closed-loop grid cell–readout–grid cell network with continuity constraint was ≈ 5×10^4. See Online Methods for parameters.

by a polynomial factor that does not qualitatively affect the exponential advantage of the GPC (**Supplementary Results**, section 4).

To summarize, the GPC enables estimation with asymptotically zero fractional error at a finite (nonzero) information rate, which CPCs cannot achieve. Furthermore, the decline in fractional error with neuron number is exponential for the GPC (inverse fractional squared error $\sim a^N$ for some constant $a > 1$) compared with the polynomial improvement (inverse fractional squared error $\sim N$ or $\sim N^2$) achieved by CPCs over identical coding ranges (**Fig. 3e**).

GPC decoding can be performed by a simple neural network

We have demonstrated, from an ideal observer perspective, the error-control properties inherent to the observed responses of grid cells. However, our analysis has been agnostic on whether the brain could or does exploit these properties. Indeed it cannot, unless it contains an appropriate decoder to infer the correct phases. We found that a relatively simple neural network architecture, consisting of grid cells projecting forward to a readout stage, recurrent global inhibition in the readout stage and symmetric back-projections from the readout stage to grid cells, can perform the appropriate decoding and correction, and thus enable accurate velocity-to-location integration in the brain.

The spatially patterned firing rates of all grid cells across networks are the inputs to the network model (Online Methods). Suppose the grid cell–to–readout weights are set by Hebbian learning on the activations of grid cells and readout cells as the animal runs through a space in the presence of spatially informative external cues. During this run, we assume that grid cell activation is without path integration errors because of the external cues, and that readout cells are separately driven to be sparsely active at one (or a few, see **Supplementary Results**, section 10) preferred location(s) (**Fig. 4a**). This sparse drive may be generated inside the readout network or by a separate pathway based on external spatial cues (for example, hippocampal place cells are driven by landmark cues[32,33] in addition to path-integrated input). After learning, each readout cell is wired to grid cells of all of the periods that are active at its preferred location(s) (**Fig. 4a**). These weights constitute templates of the correct GPC firing patterns for each readout cell's preferred location(s). The visited locations in the learning run determine R_l.

After learning, and when reliable external cues are absent, grid cells with erroneous activation resulting from noisy path integration

drive the readout cells (equation (2); see **Fig. 4b** and Online Methods). The drive to a readout cell is proportional to the match between its preferred location-specific weight templates and the present grid cell activity patterns. The maximally driven readout cell, assumed to be identified through winner-take-all or similar attractor dynamics using global inhibition[26,34,35], has a weight template best-matched to the current grid cell activity, and its preferred location is the decoded or inferred location for the noisy grid cell input (**Fig. 4b**; this is approximately equivalent to maximum likelihood estimation of the nearest codeword; see **Supplementary Results**, section 5 and 6 and ref. 26). Similar results occur when readout cells have more than one preferred location, if the readout ensemble is sufficiently distinct for each location (**Supplementary Results**, section 10).

The readout-grid cell return weights enable ongoing error correction: the winning readout cell activates the grid cells connected to it, resetting the grid cells to the state consistent with the inferred location (**Fig. 4c**, Online Methods and **Supplementary Results**, section 5 and 6). Ongoing error correction, when engaged as the animal is continually estimating its location over time in the absence of external spatial cues and with added phase noise per time step, sharply curtails the size of accumulating integration error in the grid networks. This process enables more accurate location estimation that is far more consistent with behavior than is possible with CPCs (containing as many neurons as grid cells), whose performance is too poor to guide homing behavior over long times and paths (**Fig. 4d**).

Thus far, error correction has not relied on prior information about how animal location evolves in the legitimate range. If the readout is allowed to exploit the continuity of animal trajectories in space and time, then the GPC's interleaving property can enable correction of noise much larger than $d_{\min}/2$, with the same final accuracy as when noise is smaller than $d_{\min}/2$ (**Fig. 4e**, Online Methods and **Supplementary Results**, section 9). Such a prior can be implemented by the same readout network, minimally modified; winner-take all competition in the readout is simply biased by an excitatory drive from the last winner cell to others with nearby preferred locations. With this prior, GPC location estimation remained accurate over long trajectories, even when every network phase was perturbed by ~20% of its total range per 200-ms simulation time step (**Fig. 4e**).

Indeed, the same readout network with unchanged weights can be used for correcting the GPC state on the basis of external sensory cues, if the cues drive the CA1 readout cells (data not shown). More generally, the same readout network architecture may be used to perform probabilistic inference[36], to combine variably reliable external cues with internal estimates.

DISCUSSION

Sensing and common-mode errors

Errors in velocity estimation, if common to all grid networks, or other correlated phase changes across networks cannot be corrected in the GPC. Such 'common-mode' errors are indistinguishable downstream from true animal velocity, and cannot be reduced by the GPC (or by any CPC). Thus error-correction in the GPC enables accurate estimation of the animal's trajectory, limited primarily by shared (velocity or landmark sensing) errors, rather than by neural integration noise as in conventional (CPC) neural integrators.

The GPC, if restricted to coding discrete variables (integer-valued x), can be mapped to a well-characterized error-correcting code: the non-systematic redundant residue number system[25,37,38]. The actual GPC is a generalization of this code to the analog coding domain, and as such may be of interest for analog-communications applications.

Mapping the network model to the hippocampus

The model readout maps neatly onto CA1 of the hippocampus (**Supplementary Results**, supporting figure 4), as CA1 receives direct convergent input from many dorsoventral levels of the entorhinal cortex where grid cells vary in spatial period[39,40]; it is involved in sparse location representation through place cells[41], it contains widespread inhibitory circuitry that can mediate global inhibition and, consequently, winner-take-all dynamics, and it sends its outputs back to the entorhinal cortex[39,40]. Furthermore, entorhinal-CA1 synapses display a form of Hebbian long-term potentiation similar to that assumed for setting up grid cell–readout weights[42], at least in the presence of simultaneous CA3 input to the target CA1 cells[43].

Recurrent excitatory connectivity and plasticity in CA3 make it the natural candidate for learning and representing priors. Previously active place cells can drive others cells with similar place preferences to implement the continuity prior, whereas chain-like excitatory connectivity strengthened from repeated route following can implement route-based priors (with sequence replay in CA3 being a natural consequence[44]).

Predictions

Our results can be divided into two parts, an ideal-observer analysis of the observed grid response and a network model of how it might be used. The ideal-observer results are a series of deductions about what the grid response necessarily implies about location estimation. They are analytical and, unlike numerical results, can be directly read off to generate predictions for any parameter or variable value. The model readout network is a specific, but not necessarily optimal or unique, hypothesis about how the intrinsic GPC properties may be exploited by the brain. The architecture and activity under normal and simulated lesion conditions in the model readout comprise the set of all of its predictions about connectivity and dynamics in the brain's entorninal-hippocampal network. A partial sampling of predictions follows.

First, the GPC, with hippocampus as the decoder, should perform orders-of-magnitude better idiothetic location estimation than is predicted from the dynamics of the individual GPC networks alone[24] or from the results on CPC integrators. Second, the hippocampus is essential for correcting GPC estimates based not just on external cues, as has been suggested[45], but in an ongoing manner in the absence of

external cues. With CA1-entorhinal projections disabled (but CA1 possibly intact[46]), location estimation should deteriorate in the presence and absence of landmarks; errors should greatly exceed those predicted for CPC integrators, as a result of the exquisite GPC noise sensitivity (**Fig. 1e**). Third, homing and discrimination between similar-looking locations based on idiothetic integration should be possible over distances greatly exceeding all of the grid periods, as the GPC enables accurate estimation of location over such distances. Fourth, CA1 should exhibit winner-take-all dynamics, including a race to threshold and strong subthreshold inhibition for all neurons[47]. Fifth, CA3 should drive multiple CA1 place cells with preferred locations that are nearby or parts of commonly taken paths, whereas entorhinal input should drive multiple CA1 place cells with diverse preferred locations (similar to the multi-peaked distribution of **Fig. 1c**). Entorhinal drive to CA1 should be increasingly multi-peaked, with increasing uncertainty about location. Sixth, if CA3 provides the prior (defining the candidate pool of winners in location estimation), then entorhinal input alone should not be capable of driving CA1 cells to be winners. Winner CA1 cells must be among those receiving concurrent CA3 and entorhinal input, consistent with reports on the efficacy of the two pathways to CA1 (refs. 48,49).

Readout network complexity

The readout network costs time (settling time of winner-take-all dynamics) and space (array of readout cells), both of which are measures of complexity. If the readout is unary, as in our network model, and its complexity is included in evaluating the efficiency of the GPC, but CPCs are assumed to have no readout, then the relative advantage of the GPC declines substantially (**Supplementary Results**, section 9). Indeed, readout complexity may be why the brain employs CPCs over EPCs: if fast readout is critical and representational noise is not the bottleneck (for example, when errors do not accrue), the brain may use weaker codes.

However, the generation of useful location estimates over trajectories is a difficult problem that critically depends on inference that combines noisy displacement estimates with prior expectations about location and with external landmark cues[50], regardless of how the integrator is encoded. Thus, a CPC integrator would also require a readout network similar to the GPC decoder network, capable of coding priors and doing inference. If CPCs are read out by such networks, or if the GPC readout network possesses a combinatorial location representation, then the exponential advantage of the GPC is recovered (**Supplementary Results**, section 9).

Signatures of EPCs

On the most concrete level, the GPC is an EPC because of its multi-periodic tuning curves. However, the abstract structure of the GPC coding space provides a more general view of the properties that make it an EPC: the GPC coding space is exponentially large, with codewords embedded in an interleaving arrangement. Other codes lacking multi-periodic representations may possess similar abstract properties.

Another perspective on the GPC yields a different insight: each network (unimodally) encodes a different spatial phase. Each network's phase, even if error free, is only a partial representation of location: it cannot specify location uniquely. The set of partial representations of a continuous variable is a generalization of the concept of parts-based representations for discrete variables. The different partial representations are generated from a single variable through heterogeneous tuning. Our analysis formalizes the advantages of such heterogeneity and may help to answer questions such as why the retina has multiple cells with different response properties overlapping a single retinal location, rather than multiple copies of the same cell-type.

We conclude with our aims, which were to explicate the properties of the observed grid cell location code, to determine whether the brain contains a class of EPCs capable of unprecedented error control, and to motivate further exploration of analog neural codes from the perspective of strong error-correction.

METHODS

Methods and any associated references are available in the online version of the paper at http://www.nature.com/natureneuroscience/.

Note: Supplementary information is available on the Nature Neuroscience website.

AUTHOR CONTRIBUTIONS

I.F. conceived the model. S.S. performed simulations and analyzed the data. I.F. and S.S. performed analytical calculations and wrote the manuscript.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

Published online at http://www.nature.com/natureneuroscience/.
Reprints and permissions information is available online at http://www.nature.com/reprints/index.html.

1. Softky, W.R. & Koch, C. The highly irregular firing of cortical cells is inconsistent with temporal integration of random epsps. *J. Neurosci.* **13**, 334–350 (1993).
2. Stevens, C.F. & Wang, Y. Changes in reliability of synaptic function as a mechanism for plasticity. *Nature* **371**, 704–707 (1994).
3. Seung, H.S. Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron* **40**, 1063–1073 (2003).
4. Kao, M.H., Doupe, A.J. & Brainard, M.S. Contributions of an avian basal ganglia-forebrain circuit to real-time modulation of song. *Nature* **433**, 638–643 (2005).
5. Fiete, I.R. & Seung, H.S. Gradient learning in spiking neural networks by dynamic perturbation of conductances. *Phys. Rev. Lett.* **97**, 048104 (2006).
6. Hubel, D. *Eye, Brain and Vision* (Scientific American Library, 1988).
7. Taube, J.S., Muller, R.U. & Ranck, J.B. Jr. Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. *J. Neurosci.* **10**, 420–435 (1990).
8. O'Keefe, J. & Dostrovsky, J. The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. *Brain Res.* **34**, 171–175 (1971).
9. Wilson, M.A. & McNaughton, B.L. Dynamics of the hippocampal ensemble code for space. *Science* **261**, 1055–1058 (1993).
10. Schreiner, C.E., Read, H.L. & Sutter, M.L. Modular organization of frequency integration in primary auditory cortex. *Annu. Rev. Neurosci.* **23**, 501–529 (2000).
11. Seung, H.S. & Sompolinsky, H. Simple models for reading neuronal population codes. *Proc. Natl. Acad. Sci. USA* **90**, 10749–10753 (1993).
12. Abbott, L.F. & Dayan, P. The effect of correlated variability on the accuracy of a population code. *Neural Comput.* **11**, 91–101 (1999).
13. Sompolinsky, H., Yoon, H., Kang, K. & Shamir, M. Population coding in neuronal systems with correlated noise. *Phys. Rev. E* **64**, 051904 (2001).
14. Latham, P.E., Deneve, S. & Pouget, A. Optimal computation with attractor networks. *J. Physiol. (Paris)* **97**, 683–694 (2003).
15. Brunel, N. & Nadal, J.-P. Mutualinformation, fisher information and population coding. *Neural Comput.* **10**, 1731–1757 (1998).
16. Zhang, K. & Sejnowski, T. Neuronal tuning: to sharpen or broaden? *Neural Comput.* **11**, 75–84 (1999).
17. Bethge, M., Rotermund, D. & Pawelzik, K. Optimal short-term population coding: when fisher information fails. *Neural Comput.* **14**, 2317–2351 (2002).
18. MacKay, D. *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, 2004).
19. Shannon, C. A mathematical theory of communication. *Bell Syst. Tech. J.* **27**, 379–423 623–656 (1948).
20. Goblick, T. Theoretical limitations on the transmission of data from analog sources. *IEEE Trans. Inf. Theory* **11**, 558–567 (1965).
21. Hafting, T., Fyhn, M., Molden, S., Moser, M.-B. & Moser, E. Microstructure of a spatial map in the entorhinal cortex. *Nature* **436**, 801–806 (2005).
22. Fuhs, M.C. & Touretzky, D.S. A spin glass model of path integration in rat medial entorhinal cortex. *J. Neurosci.* **26**, 4266–4276 (2006).
23. Burgess, N., Barry, C. & O'Keefe, J. An oscillatory interference model of grid cell firing. *Hippocampus* **17**, 801–812 (2007).
24. Burak, Y. & Fiete, I.R. Accurate path integration in continuous attractor network models of grid cells. *PLOS Comput. Biol.* **5**, e1000291 (2009).
25. Fiete, I.R., Burak, Y. & Brookings, T. What grid cells convey about rat location. *J. Neurosci.* **28**, 6856–6871 (2008).
26. Deneve, S., Latham, P.E. & Pouget, A. Reading population codes: a neural implementation of ideal observers. *Nat. Neurosci.* **2**, 740–745 (1999).
27. Zhang, K. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *J. Neurosci.* **16**, 2112–2126 (1996).
28. Tsodyks, M. Attractor neural network models of spatial maps in hippocampus. *Hippocampus* **9**, 481–489 (1999).
29. Xie, X., Hahnloser, R.H.R. & Seung, H.S. Double-ring network model of the head-direction system. *Phys. Rev. E* **66**, 041902 (2002).
30. Brun, V.H. *et al.* Impaired spatial representation in CA1 after lesion of direct input from entorhinal cortex. *Neuron* **57**, 290–302 (2008).
31. Paradiso, M.A. A theory for the use of visual orientation information which exploits the columnar structure of striate cortex. *Biol. Cybern.* **58**, 35–49 (1988).
32. Knierim, J.J., Kudrimoti, H.S. & McNaughton, B.L. Interactions between idiothetic cues and external landmarks in the control of place cells and head direction cells. *J. Neurophysiol.* **80**, 425–446 (1998).
33. Lever, C., Burton, S., Jeewajee, A., O'Keefe, J. & Burgess, N. Boundary vector cells in the subiculum of the hippocampal formation. *J. Neurosci.* **29**, 9771–9777 (2009).
34. Jin, D.Z. & Seung, H.S. Fast computation with spikes in a recurrent neural network. *Phys. Rev. E.* **65**, 051922 (2002).
35. Shamir, M. The temporal winner-take-all readout. *PLOS Comput. Biol.* **5**, e1000286 (2009).
36. Ma, W.J., Beck, J.M., Latham, P.E. & Pouget, A. Bayesian inference with probabilistic population codes. *Nat. Neurosci.* **9**, 1432–1438 (2006).
37. Hanzo, L., Liew, T. & Yeap, B. *Turbo Coding, Turbo Equalisation And Space-Time Coding* (Wiley-IEEE Press, 2002).
38. Sudan, M. Ideal error-correcting codes: unifying algebraic and number-theoretic algorithms. in *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes* (eds. Boztas, S. & Shparlinski, I.E.) 36–45 (Springer Berlin/Heidelberg, 2001).
39. Amaral, D.G. & Witter, M.P. The three-dimensional organization of the hippocampal formation: a review of anatomical data. *Neuroscience* **31**, 571–591 (1989).
40. van Strien, N.M., Cappaert, N.L.M. & Witter, M.P. The anatomy of memory: an interactive overview of the parahippocampal-hippocampal network. *Nat. Rev. Neurosci.* **10**, 272–282 (2009).
41. McNaughton, B.L., Barnes, C.A. & O'Keefe, J. The contributions of position, direction, and velocity to single unit activity in the hippocampus of freely-moving rats. *Exp. Brain Res.* **52**, 41–49 (1983).
42. Remondes, M. & Schuman, E.M. Molecular mechanisms contributing to long-lasting synaptic plasticity at the temporoammonic-CA1 synapse. *Learn. Mem.* **10**, 247–252 (2003).
43. Takahashi, H. & Magee, J.C. Pathway interactions and synaptic plasticity in the dendritic tuft regions of CA1 pyramidal neurons. *Neuron* **62**, 102–111 (2009).
44. Carr, M.F., Jadhav, S.P. & Frank, L.M. Hippocampal replay in the awake state: a potential substrate for memory consolidation and retrieval. *Nat. Neurosci.* **14**, 147–153 (2011).
45. Burgess, N. Grid cells and theta as oscillatory interference: theory and predictions. *Hippocampus* **18**, 1157–1174 (2008).
46. Hafting, T., Fyhn, M., Bonnevie, T., Moser, M. & Moser, E. Hippocampus-independent phase precession in entorhinal grid cells. *Nature* **453**, 1248–1252 (2008).
47. Ahmed, O.J. & Mehta, M.R. The hippocampal rate code: anatomy, physiology and theory. *Trends Neurosci.* **32**, 329–338 (2009).
48. Jarsky, T., Roxin, A., Kath, W.L. & Spruston, N. Conditional dendritic spike propagation following distal synaptic activation of hippocampal ca1 pyramidal neurons. *Nat. Neurosci.* **8**, 1667–1676 (2005).
49. Ang, C.W., Carlson, G.C. & Coulter, D.A. Hippocampal CA1 circuitry dynamically gates direct cortical inputs preferentially at theta frequencies. *J. Neurosci.* **25**, 9567–9580 (2005).
50. Ristic, B., Arulampalam, S. & Gordon, N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications* (Artech House Radar Library, 2004).

ONLINE METHODS

Network phase as a function of time and animal location. The phase of each GPC network at $x(t),t$ is given by equation (2). We assume that the accruing integration error is the dominant error in the problem. If the location $x(t)$ is perfectly known because of external sensory cues, then the integration error is zero. The vector of the N grid network phases is $\vec{\phi} = \{\phi_1,...,\phi_N\}$.

Comparable noise and coding setups for the GPC and CPC. Both the GPC and the CPC encodings involve NM neurons: N networks of M neurons each in the GPC, and one network of MN neurons in the CPC. The CPC response is characterized by a single phase variable, ϕ_{CPC}, instead of N different phase components ϕ_α as in the GPC. The spatial period corresponding to ϕ_{CPC} is R_{CPC}. Equation (2) also describes CPC phase, if the subscript α is replaced by CPC and λ_α is replaced by R_{CPC}. Thus, the CPC can be thought of as one grid network, with each neuron replaced by N neurons, and with spatial period R_{CPC}.

Phase errors of variance σ_α^2 in each grid network are compared against a CPC with phase error of variance σ_α^2 / N (**Table 1**). The $1/N$ reduction of the CPC phase squared error reflects the best-case decrease in error in a neural code when the number of neurons is increased by a factor of N with fixed coding range and tuning curve width[11–16]. The truly optimal case is to also reduce the CPC tuning curve width by $1/N$, which results in a best-case scaling of squared error of $1/N^2$ (our results are qualitatively unchanged if the CPC phase error is σ_α^2 / N^2; the exponential gains of the GPC over the CPC persist, but there is an additional factor of $1/\sqrt{N}$ in equation (6) and an additional factor of $\sqrt{N}$ in equation (7); the results are qualitatively unchanged if CPC phase error scales as some other power of $1/N$). These scalings of the CPC versus GPC phase error are assumed to make our comparisons as conservative (favoring the CPC) as possible.

Note that independent noise in the time-varying network phases, as considered here, is equivalent to independent, time-varying noise in the network periods. If all periods were to expand by the same fraction, corresponding to correlated noise across periods, all phases would shift in concert. This would result in a simple rescaling of location (distances) by a decoder ignorant of the expansion.

Spatially patterned neural firing responses. If the spatial phase of the αth network is $\phi_\alpha(x,t)$, it is equivalent to saying that (up to small shifts in the peak activation caused by the readout error) the firing rate of the ith neuron in the network is

$$r_{\alpha i}(x,t) = f_\alpha(\|\phi_\alpha(x,t) - \phi_i^*\|) \tag{8}$$

where f_α is the shape of the neural tuning curve at each preferred location in the grid cell response and ϕ_i^* is the preferred spatial phase of the ith cell. If $\phi_\alpha(x,t) = \underline{\phi}_\alpha(x(t))$ (noise-free phases, equation (1)) for all times t, the neural responses $\underline{r}_{\alpha i}(x,t) = r_{\alpha i}(\underline{\phi}_\alpha(x(t)))$, plotted as a function of x over a long time interval $[0,T]$, will be perfectly periodic, with period λ_α. If the network phases are given by equation (2), the responses $r_{\alpha i}(x,t)$ accumulated (summed) over $[0,T]$ and plotted against x will look increasingly smeared and not necessarily periodic, with increasing phase variance and integration interval length T.

Because we express the tuning curve f as a function of phase, it is unimodal (the tuning curve as a function of x would be multimodal and periodic). We model it by a Gaussian

$$f_{\sigma_e}(\phi, \phi^*) = \exp\left(-\frac{\|\phi - \phi^*\|^2}{2\sigma_e^2}\right) \tag{9}$$

where

$$\|\phi\| \equiv \min(|\phi|, 1-|\phi|) \tag{10}$$

is the distance metric on phases. The width σ_e in phase is independent of the underlying grid networks' spatial response periods. Thus, the firing field width, plotted in real space, scales linearly with the grid period. This is consistent with experimental data[21]: grid cell responses with different periods look to be uniformly scaled versions of each other. The network periods are $\lambda_1 < \lambda_2 < ... < \lambda_N$. For the CPC, we use $\sigma_p = \sigma_e$ or $\sigma_p = \sigma_e/\sqrt{N}$. The choice $\sigma_p = \sigma_e$ corresponds to equal lifetime and population sparseness for individual grid cells and CPC cells, if $R_{CPC} = R_l$. Our results do not change qualitatively for a different choice, for example, $\sigma_p = \sigma_e/\sqrt{N}$ or $\sigma_p = \sigma_e/N$ (**Supplementary Results,** section 4).

Coding range. We assume that the legitimate coding range (R_l) far exceeds any grid period ($\sim\lambda$), so that $\lambda << R_l << R$. Indeed, animals can forage 100–1,000 m away from home per linear dimension per day, eclipsing the 0.1–10-m range of grid period magnitudes. The represented ranges for the GPC and CPC are always equal, $R_{CPC} = R_l$.

Network weights, inference and correction. We describe here the network computations that occur in one pass through the error-correcting loop (assumed to take $\Delta t = 200$ ms).

The readout network infers instantaneous location by finding the maximally activated readout neuron

$$\hat{i} = \arg\max_i h_i \tag{11}$$

where $h_i(x,t)$ are the summed inputs to the readout cells. The operation to identify the maximally driven readout cell is assumed to happen in CA1 through (group) winner-take-all dynamics or other attractor dynamics which produce a narrow distribution of active cells. To mimic the result of a dynamical winner-take-all operation, the maximally driven cell becomes maximally active, and the rest are silenced, so that $r_{i \neq \hat{i}}(x,t) = 0$ and

$$r_{\hat{i}}(x,t) = 1 \tag{12}$$

The preferred location of the most active cell is the inferred location

$$\hat{x}(t) = x_i^* \tag{13}$$

The summed input h_i to the ith readout cell depends on the firing patterns of the grid cells (equation (8)) and on the learned grid cell–readout weights $W_{ij\alpha}$:

$$h_i(x,t) = \sum_{j\alpha} W_{ij\alpha} r_{\alpha j}(x,t) + h_i^0(x,t) \tag{14}$$

Here, h_i^0 represents any non–grid cell input to readout cell i, including input originating from external sensory cues, or input based on predictions or learned contextual priors.

The weights $W_{ij\alpha}$ are set once at the beginning, through Hebbian learning, in one simulated run over the range R_l with noise-free grid cell activation due to external cues and sparse place cell activation

$$W_{ij\alpha} = \sum_{x'=0}^{R_l} r_i(x') r_{\alpha j}(\underline{\phi}_\alpha(x')) \tag{15}$$

where the sum is over (sufficiently finely) sampled locations in the legitimate range. $r_{\alpha j}(\underline{\phi}_\alpha(x'))$ is the correct activity pattern of the grid cells for the location x' (that is, only grid cells with tuning curve peaks centered near x' are activated). $r_i(x')$ is the locally peaked response of the readout neurons, assumed to be driven at random or by landmark-based input during learning. We let

$$r_i(x) = G_{\sigma_h}(|x - x_i^*|) \tag{16}$$

where x_i^* designates the location of that readout cell's peak response. This is the readout cell's preferred location. In the simulations of **Figure 4**, $G_\sigma(x-\mu)$ is bell-shaped with mean μ and variance σ^2. Because the learning trajectory is only over the range R_l, readout cells are only activated and assigned preferred locations in that range.

Correction of the grid cell phases after inference (winner-take-all) in the readout network is based on the summed drive $g_{\alpha k}$ from the readout network to the grid cells, with the weights from the readout to grid cells being the transpose of the learned weights from grid cells to the readout

$$g_{\alpha k}(\hat{x}(t)) = \sum_i W_{ik\alpha} r_i(\hat{x}(t)) \tag{17}$$

The grid states activated as a result of this input drive are shown in **Figure 4c**. For the next iteration of integration and correction, the activated grid states were assumed to revert to their steady state patterns, which most resemble this input drive (share the same peak of activation). For the open-loop simulations of **Figure 4**, everything is as described above, except that the grid cell states were not corrected on the basis of the return drive from readout cells (that is, phases were not reset on the basis of $\hat{x}(t)$).

The mechanism for equations (11) and (13) can be implemented by a spiking readout network with global inhibition[34,35], which could quickly select the maximally activated readout cell while silencing the rest. Similarly, if the grid cell

networks were implemented as recurrent networks, as in references 22,24, then drive from the winning readout cell(s) through the transpose of the grid cell–readout weights W would activate the steady state grid cell pattern that is peaked around the peak of this corrective drive.

For the CPC network,

$$\hat{x}(t) = R_l \times \phi_{\text{CPC}}(t).$$

Continuity prior as an external drive to the readout cells. The contextual prior used in **Figure 4c** is the 'ball prior'

$$P(x(t) \mid \hat{x}(t - \Delta t)) \propto \begin{cases} 1 & \text{if } x(t) \in B_{v_{\max}}(\hat{x}(t - \Delta t)) \\ 0 & \text{otherwise} \end{cases}$$

Here, $B_{v_{\max}}(y)$ is the ball of radius $v_{\max}\Delta t$ centered about the location y, and $v_{\max}$ is the maximum possible speed of the animal. In one time step, the animal cannot leave $B_{v_{\max}}(y)$ if its last location was y.

This continuity constraint is easily implemented in the network model by biasing the readout's winner-take-all competition with an excitatory drive to all cells in the ball prior, using the term h_i^0. Cell i receives a current $h_i^0 \neq 0$ if its preferred location is inside the prior ball of the last decoded location, that is, if $x_i^* \in B_{v_{\max}}(\hat{x}(t - \Delta t))$. This input can be interpreted as excitatory drive from cells with similar preferred locations that were recently active.

Because this continuity prior is based on the last inferred (not necessarily true) location, it is only approximate. If the last inferred location is wrong, this approximate prior becomes a false belief.

Parameters and details for results in figures. In all simulations, $\sigma_e = 0.11$ (the full-width at half-max of the grid cell tuning curves equals 1/4 of the response period) and the preferred grid phases are distributed evenly over [0,1). Grid periods are spaced at regular intervals of 4 cm, starting from the smallest, λ_1. For better correspondence with actual neural responses, all numerical results and simulations are based on the full firing-rate response vector $\vec{r} = \{r_{\alpha i} \mid i = 1, ..., M; \alpha = 1, ..., N\}$ (equation (8)). Thus, quantities such as $d_{\min}$ are computed in the space of spatially patterned firing rates (not the informatically equivalent space of phases, which would produce qualitatively similar overall results). Phase noise in all simulations is implemented by truncating a Gaussian with zero mean and variance σ_α^2 in its tails, at $4 \times \sigma_\alpha$. This truncation is so deep in the tails that it affects only 1 in e^8 samples on average; thus, in a sample of size smaller than ~3,000, it is indistinguishable from the full Gaussian. The reason for truncation is to ensure that we are either exclusively in the correctable (noise amplitude $< d_{\min}/2$) or the uncorrectable regime.

In **Figures 1** and **3**, $\lambda_1 = 10$ cm and $M = 50$. In **Figures 1e,f** and **3b**, $N = 5$. The resulting range, estimated by increasing x in steps of size $\Delta x = 0.25$ cm, is $R = 90089.75$ cm. In **Figures 1e,f** and **3b**, $\sigma_\alpha = 0.04$. In **Figure 1f**, $R_l = 500$ cm, a value that produces $d_{\min}/2 \approx 3.87/2$, which is typically larger than the amplitude of the perturbations. Thus, the perturbed phase vectors typically map to locations outside R_l or in a narrow spread of <0.75 cm around the true location. In **Figure 1e,f**, we plotted the maximally likely locations $\hat{x}_0$ for the rate vector $\vec{r}(\vec{\phi})$ corresponding to the noisy phase $\vec{\phi} = \vec{\phi}(x_0) + \vec{\xi}$ at true animal location x_0 (explicit ML decoding), $\hat{x}_0 = \arg\min_{x \in [0, R_l]} |\vec{r}(\vec{\phi}) - \vec{r}(\vec{\phi}(x))|$, with $R_l = R$ in **Figure 1e** and $R_l << R$ in **Figure 1f**. The distance between firing rates is the Euclidean (L2) norm. Explicit ML decoding was also used in **Figure 3d,e**.

To verify that our estimate of $d_{\min}$ and R (for example, in **Fig. 3b**) was independent of the sampling grain Δx, we set Δx equal to 1, 0.5, 0.25 or 0.005 cm, and obtained essentially identical results. For convenience, we therefore sampled locations every $\Delta x = 0.25$ cm for **Figures 1** and **2**.

In **Figure 3c**, N was varied (thus R changes), but $R_l = 500$ cm was held fixed. In **Figure 3d**, $\sigma_\alpha = 0.05$, which always produced error smaller in amplitude than $d_{\min}/2$. Thus, our results are from the regime in which error correction is possible. R_l was varied to keep $\rho \approx 0.5$ (using the value $\lambda\Delta\phi = 1$ cm) as N (and thus R) increased; this criterion led to values of $R_l \approx \{118, 307, 799, 2,076, 5,394, 14,015, 36,413, 94,606\}$ cm, respectively, for the increasing values of N in the plot. The resulting exponential decay of the ratio of GPC and CPC squared errors with N at fixed ρ holds regardless of the specific estimate of $\lambda\Delta\phi$ in computing ρ. For values of $\lambda\Delta\phi < 1$ cm, the exponentially decaying curves are upper bounded by the curve in **Figure 3c**.

Figure 3e quantifies the performance of the GPC as a function of the SNR per information bit (also known as E_b/N_0) of the code. The SNR per bit is defined as the average total signal energy of a codeword divided by the average total noise energy, all divided by the number of information bits in a codeword. Thus,

$$\frac{E_b}{N_0} = \frac{\left\langle |\sum_{\alpha i} |r_{\alpha i}(\vec{\phi})|^2 \right\rangle_{\vec{\phi}}}{\left\langle |\sum_{\alpha i} |r_{\alpha i}(\vec{\phi}) - r_{\alpha i}(\vec{\phi} + \vec{\xi})|^2 \right\rangle_{\vec{\phi}, \vec{\xi}}} \frac{1}{\log\left(\frac{R_l}{\lambda\Delta\phi}\right)} \quad (18)$$

where $r_{\alpha i}$ is the firing rate of the αi neuron, and the angle brackets $\langle \ \rangle_{\vec{\phi}}, \langle \ \rangle_{\vec{\phi}, \vec{\xi}}$ denote expectations computed over all coding phases or over all coding phases and the Gaussian phase noise, respectively. In **Figure 3e**, $N = 9$ with resulting estimated $R = 29,099,069.75$ cm. We set $R_l = 5,000$ cm. Assuming $\lambda\Delta\phi = 1$ cm, this corresponds to an information rate $\rho \approx 0.5$. Assuming $\lambda\Delta\phi = 0.1$ cm, $\rho \approx 0.56$. We then computed E_b/N_0 as in equation (18) for various values of σ_α and numerically estimated the probability P_e that position is incorrectly decoded from the corresponding noisy phases (1,000 noise samples). The probability of incorrect decoding equals the probability the firing rate vector for the perturbed phase falls a distance $d_{\min}/2$ or more from the firing-rate vector for the correct phase (this is by definition of $d_{\min}$). The value of E_b/N_0 for the CPC is derived identically, but with no sum over α and with the total number of information bits in the code given by $\log(R_l/(R_{\text{CPC}}\Delta\phi_{\text{CPC}})) = \log(NM)$. The criterion for a CPC decoding error is when the decoded location falls outside a threshold distance of the true location, with the same numerical threshold as used for the corresponding GPC. The curve for the Reed-Solomon code corresponds to an information rate of 0.936 (ref. 18).

In **Figure 4d,e**, $N = 12$, $M = 50$, $\lambda_1 = 30$ cm, $\lambda_N = 74$ cm, $v_{\max} = 50$ cm s^{-1} and $\Delta t = 200$ ms. We chose $R_l = 300$ m. For all α values, $\sigma_\alpha = 0.033$ in **Figure 4d** and $\sigma_\alpha = 5 \times 0.033 = 0.165$ in **Figure 4e**. Location readout is by the network (equations (11)–(13)) using 3,000 readout cells with preferred locations spaced evenly over R_l and widths $\sigma_h = 6$ cm. This approximates ML estimation (**Supplementary Section 6**). In **Figure 4d**, $h_i^0 = 0$. In **Figure 4e**, $h_i^0 = 15,000$, an arbitrarily large value that greatly exceeds the maximal direct grid cell input, ensuring that the winner readout cell is selected from among this set. Subsequent feedback to grid cell lattices and inference of the corrected phase are described by equation (17). The CPC consists of $NM = 600$ cells, with $\sigma_p = \sigma_e/\sqrt{N} = 0.0367$ and preferred locations spaced evenly on $[0, R_l]$. All trajectory curves represent the medians over 100 simulated trajectories. The choices for the noise variance are based on the results of integration drift in detailed dynamical models of stochastic grid cell firing in continuous attractor neural networks[24]. A recurrent grid cell network model of 32×32 Poisson-spiking neurons with periodic boundary conditions and 'perfect' (nonstochastic) connectivity produced errors in position estimation with standard deviation $\sigma_{CA} = 0.0391$, close to the value used in **Figure 4d** (the larger the network, the smaller σ_{CA}; for example, $\sigma_{CA} = 0.00874$ in a 128×128 network). The σ_α used in **Figure 4e** is to accommodate possible effects of stochastic connectivity and other sources of variability not modeled in reference 24.