Navigating with distorted grid cells

Vegard Edvardsen

Department of Computer Science
NTNU – Norwegian University of Science and Technology
Trondheim, Norway
vegarded@ntnu.no

Abstract

Grid cells in the hippocampal formation are a valuable system to study both for neuroscientists and for neural network researchers, as these neurons present both a window into higher-level cognitive processes such as navigation, as well as inspiration for how to build artificial neural navigation systems. Grid cells are believed to represent an animal’s coordinates in two-dimensional space in a general fashion, useable for geometric computations by downstream neural networks, and earlier neural models have indeed shown how grid cells can be decoded for navigational purposes. However, accumulating evidence shows that grid cells are not as stable as assumed by models, but that they exhibit various geometric distortions depending on time and place. This presents a challenge to grid cell decoding models, which mainly separate into “nested” and “combinatorial” ones. Here we present a new and simplified version of a nested grid cell decoder, demonstrate that this decoder can cope with distortions, and show how this relates to a fundamental property of nested grid cell decoding. By providing positive proof that a nested decoder can navigate with distorted grid cells, we hope to inspire further neuroscientific investigation into the biological plausibility of different models for grid cell-based navigation.

Introduction

Navigation is an interdisciplinary problem area relevant to both artificial life and neuroscience; it is an important capability for situated and embodied agents, and neuroscientific research on navigation has provided a valuable window into the inner workings of higher-level cognition. Particularly in the hippocampal formation of mammals, numerous spatially selective neurons have so far been identified (Moser et al., 2017), including place cells, border cells, head-direction cells and grid cells. Grid cells are particularly intriguing because of the highly fascinating patterns these neurons generate across space (Hafting et al., 2005), that appear almost algorithmic in nature. Specifically, they respond to the animal’s location in the two-dimensional plane in a hexagonal pattern; the neurons activate whenever the agent is located on the vertices of an equilateral triangular lattice imagined to span the entire space available to the animal (Fig. 1A).

The mathematically appealing patterns of grid cells have suggested they might implement a spatial coordinate system: The collective activity of all grid cells might uniquely describe the animal’s current coordinates and be useful for inferring geometric relationships (Fiete et al., 2008), and grid cells have now been shown to be decodable into goal vectors by hypothetical downstream neural networks, enabling an agent to navigate toward arbitrary goal locations by reading out the current activity of the grid cell population and comparing this to a stored version of the grid cell activity corresponding to the goal (Edvardsen, 2015; Bush et al., 2015). While these models are interesting architectures for artificial agents, they also provide proofs of concept that real grid cells might contribute to “vector navigation”—thus guiding future neuroscientific research into the function of grid cells.

However, the universal decodability of grid cells has been questioned, particularly by recent discoveries that grid patterns are not as stable and predictable as models have assumed (Krupic et al., 2015; Carpenter and Barry, 2016). Grid patterns have been shown to distort in various ways, for example by shearing and rescaling as a function of the amount of time spent in a particular environment. As such distortions challenge the assumptions of different proposed grid cell decoders, they might help discern any differences in their biological viability. In particular, two major approaches to grid cell decoding are the “nested” (Stemmler et al., 2015) and the “combinatorial” (Fiete et al., 2008) decoders. In this paper we show that a nested decoder can cope with distorted grid cells, without addressing the question of whether a combinatorial decoder could similarly be made able to handle the distortions. Thus, while our results allay the criticism that grid distortions might present a problem for grid cell decoding in general, they leave open the question of whether combinatorial decoding remains viable.

Next in this paper we first describe grid cells, grid modules, how to decode one module and how to combine multiple modules in a nested fashion to navigate long distances. This presents a new, simplified version of our earlier model for nested decoding (Edvardsen, 2015) that makes it easier to reason about the effects of distortions. We then add distortions to the model, show that navigation with one module still succeeds, and demonstrate that navigation with mul-
tiple, differently distorted modules also succeeds. Before concluding, we show how our results relate to a fundamental property of nested decoders, by demonstrating that the model can cope even with “perturbed” grid patterns.

**Grid cells organized into grid modules**

Fig. 1A shows examples of four different idealized grid cells, each box a top-down heatmap of a 4 m wide square arena, showing where that particular grid cell might be active. All grid cells in this paper were generated by evaluating

\[
GC(x, y) = \max \left[ 0, -0.2 + \prod_{d=0}^{2} \left( 1 + \cos \left( (x - Ox) \cdot 2\pi / S \cdot \cos (R + d \cdot \pi / 3) \cdot 2 / \sqrt{3} + (y - Oy) \cdot 2\pi / S \cdot \sin (R + d \cdot \pi / 3) \cdot 2 / \sqrt{3} \right) \right) \right]
\]

(1)

at any given \((x, y)\) location, making a hexagonal pattern by intersecting three waves 60° apart (Solstad et al., 2006). A grid cell is here characterized by the parameters \(S, R\) and \(O\); \(S\) is the scale of its pattern, given as the distance between two peaks, \(R\) is its orientation, fixed at \(\pi / 2\) in this paper, and \(O\) is the two-dimensional offset of the pattern from some arbitrary point of reference. Cells with the same scale and orientation are said to belong to a grid module, which is our fundamental unit for grid cell-based navigation; assuming there is sufficient coverage of cells in a given module, there will always be an active subset of neurons to help localize the animal no matter where it might be located. As the first three grid cells in Fig. 1A have the same scale (0.9 m) and orientation, they belong to the same module, while the fourth cell, of different scale (0.6 m), belongs to a different one.

The collective activity of all grid cells in a given module can be visualized by arranging them in a matrix according to their individual offset values \(O_x\) and \(O_y\), so that cells with similar offsets—thus highly overlapping grid patterns—are located nearby in the matrix, while cells with less overlapping patterns are located farther apart. Fig. 1B shows examples of such “neural sheet” matrices at various snapshots in time, each pixel showing the instantaneous firing rate of a particular grid cell, with all pixels in a given matrix recorded simultaneously. The first matrix shows a grid module that includes the first three cells from Fig. 1A, recorded while the agent is located at the goal in the box center. The grid cells from Fig. 1A are in this matrix represented respectively by the 1st, 6th and 11th upper row pixels from the left.

Each matrix reveals a single “bump” of activity distributed among the grid cells, potentially wrapping around horizontally from the upper left corner to the upper right corner and vertically from both upper corners to the lower middle of the matrix. This is because the neural sheet reflects the “twisted torus” topology inherent to the hexagonal grid cell pattern (Guanella et al., 2007). The second matrix shows the same module, but with the agent located 20 cm east of the goal location in box center. First three grid cells have scale 0.9 m, y offset \((O_y)\) of 0 cm and x offsets \((O_x)\) of 0 cm, 15 cm and 30 cm. Fourth cell has scale 0.6 m. (B) Matrices with snapshots of concurrent activity within a full grid module at different points in time. Each plot shows collective activity of all 30 x 26 cells in the module, each pixel indicating a given neuron’s instantaneous firing rate and arranged according to its offset values \(O_x\) and \(O_y\). First matrix shows case with agent in box center, second matrix after moving 20 cm east, third matrix another 20 cm east and fourth matrix after then moving 20 cm south. (C) Mechanism for decoding a single grid module, consisting of 12 units each responsible for detecting the expected module state should the goal be located ~8 neurons in a particular direction from the current location, here configured to navigate back to the location indicated in top matrix in B. Center matrix shows current module state. Radial bars show activation of each detector. Black semi-circular notch shows final goal direction from the population vector average of all detectors’ contributions, correctly indicating the goal to be to the north-west. (D) Trajectories from navigating with decoder outlined above. 400 trials were started from locations randomly sampled across the area of a 0.6 m radius disc. Trials terminating within 4.5 cm of true goal location were deemed successful and shown in blue. Failed trials shown in red. Trajectories were plotted with 5× subsampling throughout paper. (E) Trace of boundary where decoded vector flips from pointing inward to pointing outward (blue line). Dashed line shows maximum possible starting distance for trials in D, showing that some trials might start outside this boundary and thus fail.
previous location—the activity bump has thus shifted to the right in this updated snapshot. The third matrix shows the activity a further 20 cm east, while the fourth matrix shows the activity after the agent has then moved 20 cm south.

Navigating with a single grid module

Comparing the first and the fourth matrices, we see that the collective activity of the grid module has updated to reflect the agent’s new location southeast of the goal, by moving the bump to the right and down from its initial location in the matrix. The principle behind the grid cell decoder is to recognize the correct shift of the bump needed to bring it back to its correct place in the matrix. Fig. 1C shows how we propose to do this: A set of detectors is instantiated for 12 different directions, each detector consisting of a “template” of the expected state of the grid module should the goal be located in a particular direction away from the agent.

For a given current state of the grid cell module, the task of the decoder is then to compare that current state to each of these 12 template states. We have implemented this as the dot product between the template matrix and the current matrix (after flattening both to vectors), essentially corresponding to a neuron with multiplicative synapses between input signals from corresponding current state and target state grid cells. Fig. 1C shows how the different detectors activate in a situation corresponding to the fourth matrix in Fig. 1B—the detectors corresponding to northwesterly goal directions are the most strongly activated, thus correctly compelling the agent to move northwest in order to reach the goal.

In Fig. 1D we use this mechanism to navigate, starting the agent at random locations sampled from the area of a 0.6 m radius disc. In 1 cm timesteps, the agent updates the grid cell population to reflect its current location, compares this to the grid cell population for the goal using the mechanism outlined above, and calculates its next movement direction from the population vector average of a set of unit vectors pointing in the directions indicated by the 12 detector units, weighted by their corresponding activation values. This process iterates until the agent’s current location is less than 2.5 cm from five timesteps ago, signalling that the agent has reached what it believes to be the goal and started oscillating around this point. Fig. 1D shows the outcome of 400 such trials, blue lines indicating successful trials that ended within 4.5 cm of the goal and red lines showing failed trials.

These trials clearly separate into two classes: the successful ones that precisely hit the intended goal location, and the unsuccessful ones that instead aim directly toward one of a discrete set of erroneous target locations, distributed in a hexagonal pattern. Because grid cells have a repeating pattern, such ambiguity will inevitably arise as you go too far away from the goal. Fig. 1E clarifies this by tracing out the boundary at which the decoded goal vector flips from pointing inward, i.e. less than 90 degrees away from the true goal direction, to instead pointing away from the goal (blue line). Considering the radius of starting locations (dashed line), some trials in Fig. 1D may evidently have started outside of the valid navigational range for this single grid module.

While the range of this system is quite short, the system is able to pinpoint the correct goal location with high precision; one immediate proposal for how to increase the navigational range of this system would thus be to trade some of this precision for an increase in range. We could e.g. zoom up the scale of the grid module by a factor of a thousand, and assuming the agent remained able to hit the goal location, we would then have gained a thousandfold increase in range “for free”. However, we cannot expect this to be viable biologically, as there will inevitably be noise in the system e.g. due to error from path integration processes (proposed to participate in maintaining grid patterns in the real brain).

Hence, for a more realistic take on the grid cell decoding problem, we explicitly introduce some jitter into the coordinates represented by each of the grid modules: specifically, on each new trial, the true goal coordinates are offset by a noise vector with random direction and with magnitude sampled from a Gaussian distribution with standard deviation 5 % of the grid scale. Once such jitter is taken into account, simply scaling up one grid module might no longer be a viable way of increasing navigational range while maintaining sufficient precision. A possible solution, for which there is biological backing, is then to combine information from multiple modules of different scales—described next.

Combining multiple grid modules for nested grid cell decoding across longer distances

In a nested view of the grid system (Stemmler et al., 2015), we assume there exists a grid module with a sufficiently long navigational range to exceed the agent’s behavioral requirements (Fig. 2A). While this module might be too jittery to successfully hit the goal location, we then assume that additional smaller-scaled modules can fine-tune the navigational resolution once the agent gets closer to the goal. Fig. 2B shows a smaller grid module (yellow) nested within a larger one (blue), as well as the criterion we propose for handing off control from the larger to the smaller module: An extra detector unit, configured to detect the target location itself, is added to the larger-scaled module—the green line here showing the area within which this unit will fire above a certain threshold. We propose that decoding is an iterative process, where the largest-scaled module has control of the decoded goal direction until its target detector exceeds its threshold. It will then yield control of the navigation process to the next, smaller module—with its own set of detectors—which assumes control until its respective target detector is sufficiently strongly activated, and so on. This nested process, executed on every timestep, can be maintained for as long as there are smaller-scaled grid modules available.

A high amount of jitter in large-scaled modules is thus acceptable as long as the agent can be delivered into the


valid navigational range of smaller-scaled modules with less debilitating jitter (Fig. 2C). Grid cells in the real brain are known to cluster into multiple coexisting grid modules of increasing scale, and intriguingly, these modules seem to have a fixed ratio between their successive grid scales, so that the sequence of grid scales within an animal constitutes a geometric progression (Stensola et al., 2012). There is thus an exponential growth in range as more modules are added.

Say we wanted to navigate from distances of up to 2 m while coping with jitter—we could e.g. configure our agent to utilize eight distinct grid modules, with grid scales progressively increasing from a lower value of 0.3 m with a fixed ratio of 1.5 between modules, for a largest scale of ~5 m. Fig. 2D first demonstrates how each of these modules would perform individually, each showing 400 trials with only one of the eight modules present. Using only a small-scaled module, trials terminate early in a multitude of erroneous goal locations. As scale increases, the “catchment area” for the correct goal location grows too, thus improving the number of trials that head toward the correct target—however, the amount of jitter increases as well, so the precision is no longer sufficient for trials to succeed. The largest grid module’s catchment area encompasses all trials, yet 87% of trials still failed because of the jitter (Fig. 2E).

If we instead combine several modules as described above, the agent should enjoy both the long range of the larger modules and the precision of the smaller modules. In Fig. 2F, with the agent now combining all eight modules in a nested fashion, almost all trials successfully reached the goal within specified tolerances. Drilling further into the agent’s behavior, histograms in Fig. 2G–H show respectively the deviation in decoded goal direction and the identity of the currently driving grid module, as a function of distance, for all timesteps across all successful trials in Fig. 2F. At far distances, the larger modules are in control of the agent—gradually yielding control to the smaller modules as the goal is approached (Fig. 2H). This maintains the deviation at a low level until the agent is immediately adjacent to the goal (Fig. 2G), thus ensuring successful navigation in these trials.

Navigating with a single distorted grid module

In the preceding sections we assumed perfectly hexagonal grid patterns, reliably adhering to the equilateral lattice everywhere. However, this is not an accurate description of the real situation, as grid patterns in real animals have been shown to distort from the perfect hexagonal grid in various ways. For example, Stensola et al. (2015) showed that with increased exposure to a particular enclosure, grid patterns may gradually shear away from the walls of the box, while Barry et al. (2007) showed that when a familiar enclosure is compressed, grid patterns may also rescale to match the new dimensions of the box. Interestingly, other co-recorded grid modules might stay fixed at their previous scales (Stensola et al., 2012), highlighting the notion that different grid

![Figure 2: Navigating with multiple grid modules. (A) Maximum trial distance was increased to 2 m (dashed line). Catchment area of largest-scaled grid module (#8) is sufficiently large to contain all trials (blue line). Thin solid lines indicate jitter in the corresponding module. (B) Nested within module #8 is the next-largest-scaled module (#7, yellow line). Drop-off area from module #8 (green line) is inside catchment area of #7, thus nested navigation should succeed. (C) In total eight modules are nested within each other, grid scales organized in a geometric progression from a smallest value of 0.3 m with a fixed ratio of 1.5 between successive modules. (D) Trial outcomes if the agent were to navigate using only one of the eight possible modules. Upper left shows 400 trials with smallest module. Module scale increases in standard reading order. Smaller-scaled modules are more ambiguous and have a denser distribution of erroneous goal locations, but the larger-scaled modules experience more jitter and hence are also insufficient by themselves. Each box shows a 4.5 m wide square top-down view of environment. (E) Failure rates in each of the situations shown in D, red indicating trials that failed outside of the expected catchment area for a module of the given scale (calculated according to 0.15 m−1.5−i for module i), yellow indicating trials that failed within the catchment area but due to jitter were not able to hit within the success criterion of 4.5 cm. (F) Trials combining all eight modules in a nested fashion. Almost all trials were successful. (G) 2D histogram showing deviation in the decoded goal direction as a function of distance (normalized per column). Histogram includes samples from every 1 cm timestep in all successful trials in F. Decoding error is kept low until immediately adjacent to the goal. (H) Histogram as in G, but here showing which module was active at the given timestep.
modules might distort incoherently. Several more examples of grid cells deviating from the idealized hexagonal pattern have been identified (Carpenter and Barry, 2016).

While the purpose and underlying mechanisms of these distortions are still unknown, their mere presence poses a challenge for grid cell navigation models: Can these decoders still function when grids are not perfectly hexagonal (Krupic et al., 2015; Carpenter and Barry, 2016)? To investigate how our decoder copes with distortions, we have thus incorporated one-axis stretching (Barry et al., 2007; Stensola et al., 2012), one-/two-axis shearing and symmetric shearing (Stensola et al., 2015) into our model as follows. Each distortion is represented in Fig. 3A by a function $D : (x, y) \mapsto (\tilde{x}, \tilde{y})$ that maps the agent’s true $(x, y)$ coordinates to the distorted coordinates $(\tilde{x}, \tilde{y})$. These distorted $\tilde{x}$ and $\tilde{y}$ values are then used when determining grid cell activation using Eq. 1. The specific expressions used for stretch, one-/two-axis shear and symmetric shear distortions are given respectively by $D_{\text{stretch}}$, $D_{\text{shear}}$ and $D_{\text{symmetric}}$ (the latter two based on formulations in Stensola et al., 2015):

$$D_{\text{stretch}}(x, y) = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}$$  

$$D_{\text{shear}}(x, y) = \begin{bmatrix} 1 & a \\ b & 1 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$D_{\text{symmetric}}(x, y) = \left( \frac{x}{1 + a} \right) \begin{bmatrix} 1 + a \cdot y / (1 + a) \end{bmatrix}.$$

See Fig. 3B for examples of resulting distorted grid cells.

The first question to address is whether navigation with a single grid module can still succeed when the constituent grid cells are distorted. If we assume that all grid cells within a given module distort in the same way, which seems to be the case biologically (Stensola et al., 2012, 2015), we expect the neural sheet to look the same as without distortions: Because distortion is a function of location, all cells will experience the same $x$ and $y$ distortion simultaneously—the activity bump in the neural sheet should thus retain its shape.

We can therefore attempt to navigate with a distorted grid module by using the same decoder as earlier, unmodified. Fig. 3C shows catchment areas traced as before, but with distortions in effect. Inside these boundaries, the decoded goal vectors point inward toward the goal, a promising sign that navigation with distorted modules might succeed. Fig. 3D presents a set of navigation trials performed with a sheared grid module as shown in Fig. 3A second column, demonstrating that a large proportion of trials remained able to reach the goal. Observe that the shape of the catchment area has been distorted as in Fig. 3C, and that the distribution of erroneous goal locations has also similarly been distorted.

Careful observation of the trajectories reveals that they might now curve; because the decoder is unaware of any distortions in effect, it would not know e.g. that a decoded goal vector pointing east might in fact correspond to a north-easterly bearing under the distorted regime. Fig. 3F shows a scatter plot of this deviation from the true goal direction during successful trials in Fig. 3D (yellow dots), revealing a clear pattern of deviation. For grid distortions that produce constant $\tilde{x}$ and $\tilde{y}$ gradients throughout space, which includes $D_{\text{stretch}}$ and $D_{\text{shear}}$ but not $D_{\text{symmetric}}$, there is a simple remedy in adjusting the decoded goal vector $(\tilde{v}_x, \tilde{v}_y)$ using

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \partial \tilde{y} / \partial x \\ \partial \tilde{x} / \partial y \end{bmatrix}^{-1} \begin{bmatrix} \tilde{v}_x \\ \tilde{v}_y \end{bmatrix}$$

as the final step. Fig. 3E shows a new set of trials employing
Navigating with multiple distorted modules

Having verified that navigation with a single grid module succeeds despite distortions, we now ask whether navigation over longer distances, combining multiple nested modules, can also succeed with distorted modules. We expect this to work as long as larger-scaled modules remain able to deliver the agent into the catchment area of the smaller-scaled ones. To investigate this we next ran a set of trials as in Fig. 2F, but with the eight modules distorted using a mix of one-axis stretching, one-/two-axis shearing and symmetric shearing (Fig. 4A, details in caption). In other words, we have configured the agent not only to utilize distorted grid modules, but to have differently distorted modules—should this succeed, it would be a stronger demonstration of capability than having one single distortion apply equally to all modules.

Example grid cells from each of the eight modules are shown in Fig. 4A, while Fig. 4B shows outlines of each module’s catchment area together with the “drop-off areas” from their respective larger-scaled modules. The condition for successful navigation is that the agent should always be dropped off fully inside the catchment area for the lower-scaled module; Fig. 4B confirms this to apply to the present configuration, but note that in some areas, the extra trial-by-trial jitter in each module’s represented coordinates might cause some of these boundaries to intersect and potentially cause problems. Fig. 4C shows 400 trials with these distorted grid modules, but without any additional jitter; all trials were successful. Navigation is still largely successful when the trials are run with additional trial-by-trial jitter (Fig. 4D), with a few exceptions due to failures from the extra jitter. Except for this caveat, though, we see that grid distortions pose no problem for the navigation process with nested grid cell decoding, and that this applies even when different modules are distorted incoherently.

Navigating with perturbed grid cells

The previous results highlight an appealing feature of nested grid cell decoders, in that they don’t require precisely interlocked conjunctions of grid activity across all modules in order to work. This is in contrast to combinatorial decoders—the main alternative to nested decoders—which exploit the combinatorial growth in the number of unique conjunctions of activity across all modules as new modules are added, to theoretically enable navigation far beyond the range of the largest module (Fiete et al., 2008). A nested decoder, on the other hand, can only navigate within the range of the largest module—although with a geometric progression of grid scales, this range does rise exponentially. In exchange, though, nested decoding faces a far simpler requirement for successful navigation, namely that larger-scaled modules need only deliver the agent into the realm of a smaller-scaled module. The smaller modules do not even need to participate in the process until the larger modules finish their jobs.

This should mean that a nested decoder can function in yet more adverse conditions than those considered so far.
Specifically, we will consider what happens to the navigational system when modules are perturbed at locations far away from the goal. By “perturbed” we mean that the cells no longer keep perfect track of the agent’s exact position relative to the goal, but that e.g. due to noisy path integration, the grid cells’ firing locations might jump around and no longer resemble grid patterns when far away from the goal.

Fig. 5A shows how we have implemented these perturbations, by using the same formalism of a \( D : (x, y) \to (\tilde{x}, \tilde{y}) \) function. However, this function no longer changes gently across space as in Fig. 3, but instead presents noisy maps that might violently jumble the grid pattern around. Specifically, the \( x \) and \( y \) offset maps were generated by selecting 300 random points within a 4 m radius disc, for each point sampling an offset value from a Gaussian distribution with standard deviation 15 cm, and then interpolating across these points (using Clough-Tocher interpolation from the SciPy software package). As mentioned above, though, we only want these perturbations to be in effect when the agent is far away from the goal—we assume that an agent would realistically be able to gradually correct these perturbations from sensory inputs as it approaches a familiar goal location. We therefore fade away the effect of perturbation toward the goal, using a Gaussian fade with standard deviation 0.6 m (Fig. 5B).

In Fig. 5C–D, to see the effects of the perturbation in more detail, we analyze a grid pattern before and after by performing two-dimensional autocorrelation on each \( 3 \times 3 \) subdivision of the spatial response heatmap. Autocorrelograms are frequently used in the neuroscience literature to quantify the degree of “gridness” of a particular neuron, as the hexagonal symmetry of a grid cell becomes apparent when correlating the spatial response map in this fashion. Fig. 5C shows that, as expected, there is hexagonal symmetry in the autocorrelograms for the non-perturbed grid cell. In Fig. 5D, however, we see that the neuron only has hexagonal autocorrelation structure in the central subdivision, i.e. close to the goal. The neuron therefore behaves as a grid cell when close to the goal, but at farther locations the perturbation might mean that this neuron would not have been classified as a grid cell.

Thus, in some respects, these perturbed cells do not behave as grid cells when sufficiently far away from the goal location. Can these neurons still contribute to a successful navigational process? Fig. 5E–H present results similarly as in Fig. 4A–D, but with a new configuration of the model where the grid modules experience a perturbation as just described. Similar to how the trial-by-trial jitter was specified as a proportion of grid scale, we find it reasonable to assume that larger-scaled modules get perturbed on a proportionally larger scale—noise points are therefore more spread-out, but have a higher amplitude, for larger-scaled grids, and the radius for the fade-away near the goal is also larger for the larger-scaled modules. Fig. 5F shows that the catchment areas for each individual module has largely retained its shape from the undistorted situation—this is as expected, since we
have assumed that the perturbation fades off toward the goal location. The conditions are therefore in place for nested decoding to succeed: when trials are run without or with jitter, shown respectively in Fig. 5G and Fig. 5H, we see that navigation with perturbed grids indeed was successful.

Discussion

In this paper we have presented a new and simplified version of our earlier nested grid cell decoder (Edvardsen, 2015), able to perform vector navigation over arbitrarily long distances using multiple modules. Our earlier model performs nested decoding as part of a larger neural navigation system that also includes path integration components to generate the grid cells; here we instead look at the decoding problem in isolation, thus we calculate the grid patterns directly for a simpler overall model. The decoding mechanism itself has also been simplified, now only requiring a set of 13 detector units per module. Whereas the earlier model decoded all modules in parallel and used different weighting of output from different modules, the new version uses a simpler approach to combining multiple modules, by making explicit the underlying principle that a nested decoder only strictly needs to follow the signal from one module at a time.

We then showed that this nested grid cell decoder can cope with distorted grid cells, and that navigation can succeed even when different grid modules experience different distortions. In Stemmler et al. (2015), which presents a nested decoder based on recursive population vector readout of the full grid cell population, they point out that it is possible to accommodate grid modules with distortions such as shearing in their decoder. However, while they compensate for these distortions directly in their readout mechanism, here we show that the decoder might not even need to be aware of the distortions for navigation to succeed.

We showed how this relates to a fundamental property of the nested view of the grid system, namely that navigation can proceed as long as each grid module can guide the agent from inside the drop-off area from a larger module and into the realm of a smaller module. We underscored this by showing that nested decoding succeeds even when grid patterns are perturbed—to the point of no longer resembling grid cells—in locations away from the goal. Our nested navigation model was able to cope with these challenges with no change to its decoding mechanism, whereas e.g. a combinatorial system sensitive to unique conjunctions of activity across all modules, would likely have to be retrained or accounted for the distortions through some other means.

We have thus demonstrated a nested grid cell decoder robust to disturbances in the form of jitter, distortions and perturbations of grid patterns. Noisy path integration can hence support successful grid decoding, so this might be a viable architecture for neuromorphic robot navigation if extended with circuitry for motor control, obstacle avoidance, etc.

There is also the question of how these results relate back to the biological system. It remains an open question whether real grid cells participate in vector navigation, and if so, whether this occurs according to nested decoding, combinatorial decoding or some other mechanism. Our results here show that a nested decoder can be robust to the challenges of noise and distortions that are relevant in a biological setting, and also that nested decoding can have modest resource requirements in terms of numbers of neurons and synapses. To address these questions on the role of real grid cells in navigation, more data is needed e.g. about how these neurons behave when animals navigate across long distances in their natural environments (Geva-Sagiv et al., 2015).

References


