ChatGPT

Continuous Attractor Networks and Toroidal Manifolds in Navigation and UI Design

Continuous Attractor Models and Toroidal Representations (Neuroscience Context)

Continuous attractor networks (CANs) are neural network models where activity can move continuously across a low-dimensional manifold of states. In the brain's navigation system, **grid cells** in the medial entorhinal cortex provide a striking example: their joint population activity lies on a two-dimensional torus, effectively forming a *doughnut-shaped* attractor manifold [1]. As an animal moves, a localized "bump" of neural activity shifts across this toroidal manifold, encoding the animal's 2D position in the environment [1] [2]. Likewise, **head direction cells** (encoding facing direction) form a one-dimensional ring attractor – a circular manifold where each point corresponds to an orientation angle [2] [3]. These brain findings validate continuous attractor models: grid cell networks maintain a stable hexagonal firing pattern independent of sensory input, as predicted if neurons are linked via recurrent excitation/inhibition on a torus [1] [2]. In short, the brain's "GPS" uses continuous attractors with periodic boundaries (circles and tori) to seamlessly represent spatial variables (position and angles) without edge effects.

Engineering Applications in Navigation (Robotics, VR, and AR)

Engineers have leveraged these principles in computational systems for navigation, including robotics and virtual reality. **RatSLAM**, a biologically inspired SLAM system, is a prime example: it employs a *3D continuous attractor network* of "pose cells" to represent a robot's location (x, y) and orientation (θ) [4]. Crucially, the network has *wrap-around connections at its edges*, effectively giving it a toroidal topology so that movement off one boundary enters from the opposite side [5]. This design (inspired by continuous attractor models of place/grid cells) lets RatSLAM maintain a persistent pose estimate as the robot moves, and even entertain multiple pose hypotheses in ambiguity [6] [7]. Milford et al. demonstrated that RatSLAM's internal cells exhibit grid-cell-like firing and can resolve navigational uncertainty by keeping multiple activity bumps (possible locations) on its attractor manifold [6] [7]. In practice, RatSLAM and its open-source implementations (e.g. OpenRatSLAM) have shown robust long-term mapping, validating the continuous attractor approach in real robots.

Modern **VR/AR systems** similarly benefit from toroidal state representations. For instance, maintaining a user's **head orientation** is naturally done on a ring: many SLAM algorithms (and even neuromorphic chips) treat heading as a circular continuous variable. A recent neuromorphic SLAM accelerator, *NeuroSLAM*, uses on-chip oscillatory networks as pose cells and a digital head-direction cell to mimic the rodent hippocampal system [8] [9]. The oscillatory network's continuous attractor property enables efficient spatial cognition with minimal power [8] [9]. By updating the activity bump based on inertial rotation input, such a system keeps track of 360° orientation without discontinuity – ideal for AR/VR devices that track headset or phone rotation. In essence, representing pose on toroidal manifolds (circles for angles, torus for 2D position) prevents "edge" glitches (like abrupt jumps at 359°->0°) and allows smooth wrap-around updates, much like the brain's head-direction ring attractor [2] .

Deep learning research has also incorporated grid-cell continuous attractors for navigation. A notable example is the DeepMind *grid-cell model*, where a recurrent neural network (LSTM) was trained to perform path integration in a simulated 2D arena [10] . Intriguingly, the network *spontaneously developed* units with hexagonal grid-like firing patterns, plus complementary head-direction cells [11] . This emergent torus-like coding gave the agent a kind of internal map, which it then leveraged to navigate to goals in unfamiliar virtual environments [11] . In effect, the deep network rediscovered the continuous attractor solution: using a smooth manifold of latent states (phase-shifted grid patterns) to encode location. Agents endowed with these grid-like representations achieved superior navigation performance, even outperforming human players in certain virtual mazes [12] . The takeaway is that continuous attractor representations (whether hand-crafted in a SLAM system or learned by a neural net) provide a stable metric space for self-position, enabling reliable path integration and vector navigation [12] .

Some systems explicitly exploit **toroidal manifolds for pose** beyond 2D navigation. In computer vision, for example, Elgammal et al. (2006) modeled the two continuous factors of human pose estimation – body configuration and viewpoint – as a point on a 2D torus [13] . By embedding "different poses × different views" on a toroidal manifold, their model could infer both 3D body posture and camera angle simultaneously from a silhouette [13] . This clever use of a torus (product of two circles) treats each cyclic variable properly and allows smooth interpolation across all combinations of pose and angle. It's a practical demonstration that when a problem involves periodic state dimensions (like rotation or phase), mapping it onto a torus can greatly simplify learning and interpolation. Similarly, robotics researchers often represent robot joint angles or orientations on circular manifolds (using sine/cosine or continuous attractors) to avoid singularities in control and planning.

UI State Mapping and Spatial Interfaces

Beyond physical navigation, continuous attractor and toroidal mapping concepts are inspiring new approaches to **user interface (UI) navigation and state management**. Just as grid cells form a spatial map of an environment, one can imagine a **"cognitive map" of UI states** where each screen or context is a location in a conceptual space. In fact, neuroscientists suspect that grid-cell networks support abstract cognition and memory organization, not just literal space [14] . For example, a grid-like code might map conceptual or task spaces in the brain, allowing us to navigate ideas or timelines as if moving through a mental landscape [14] . This suggests that UIs could leverage spatial metaphors under the hood: arranging app states on a continuous 2D manifold (even a torus) so that transitioning between related views is like traveling a short distance on a map.

In **spatial interface design**, developers already create intuitive experiences by exploiting our innate spatial memory. VR and AR interfaces often pin virtual panels or menus to fixed positions in the user's surroundings, effectively turning UI selection into a spatial navigation task. For instance, a VR toolkit might place tool palettes in a circle around the user – turning your head to the right might always reveal the "menu" panel, to the left the "chat" panel, etc. Such a layout can be thought of as mapping UI modes onto a ring around the user (a 1D circular manifold). The user's pose (head direction) directly picks the UI state, leveraging the brain's head-direction system. Because the arrangement is circular, the user can spin through options infinitely in either direction – a design that mirrors a toroidal continuum. This *360° menu* concept has appeared in some VR UIs and games (e.g. radial menus or wrap-around selection screens), providing **seamless navigation through interface options** with no hard stops at an "end" of a list.

On mobile and web apps, designers employ spatial metaphors like **carousel** or **infinite scroll** interfaces, which implicitly use a toroidal model. For example, a horizontally scrolling carousel of content can be made to wrap from the last item back to the first, forming a loop. Internally, treating the index of the carousel modulo N (number of items) is equivalent to using a circle manifold for the state. This ensures that swipe gestures can always continue moving items in a circle, creating an "endless" UI experience. Some smartphone launchers use this principle to allow cycling through home screens in one direction indefinitely. While these implementations may not cite neuroscience, they resonate with the idea of a continuous attractor – the interface state moves continuously through a ring of possibilities, rather than jumping back and forth with limits.

Researchers and futurists are actively bridging **neuroscience and UX design** in this area. It's been proposed that UIs which maintain *spatial consistency* (elements that stay in fixed spatial locations across interactions) can harness the brain's spatial memory for faster, more intuitive use [15]. For instance, a context-aware AR interface could lay out information in a persistent 3D arrangement around the user, so returning to a tool is like mentally "walking" to its location. Some have even speculated about embedding grid-cell-like codes in software: using a hexagonal grid system as an underlying coordinate frame for UI contexts, so that the distances and directions between UI states are represented in a consistent metric space (enabling the system to predict likely user transitions as vectors, akin to path integration in a cognitive map). While these ideas are in early stages, they align with evidence that grid cells can map non-physical domains (like abstract task spaces or time) with the same hexagonal geometry [14]. In other words, the **UI of the future** might internally treat your app navigation like moving through a virtual landscape – each menu, page, or AR object anchored at a point on a notional map or torus, and your navigation actions update your "coordinates" on this map.

Examples and Implementations:

- **RatSLAM (2000s)** – A robot navigation system that uses a 3D continuous attractor network to represent pose (x, y, θ). The network is implemented with toroidal (wrap-around) connectivity, and the cells exhibit grid-cell-like multiple firing fields [4] [5]. RatSLAM's open-source versions have been used in long-term robotic mapping and even in place recognition tasks using visual inputs.
- **NeuroSLAM Chip (2021)** – A custom CMOS accelerator for SLAM that mimics place and head-direction cells. It uses an oscillator-based attractor network (for location) and a digital ring attractor (for orientation) to achieve efficient spatial mapping on AR/VR edge devices [8] [9].
- **DeepMind Grid Navigation (2018)** – A deep RL agent whose recurrent neural network developed *grid-like units* after training on path integration [11]. This provided a torus-like internal map that the agent used for goal-directed navigation, demonstrating the power of continuous attractor representations in software agents.
- **Torus Manifold Pose Inference (2006)** – In a human pose estimation model, two periodic variables (body pose phase and viewpoint angle) were embedded in a 2D torus manifold for learning. The system mapped input images to a point on the torus, enabling simultaneous recovery of pose and view with a smooth continuous latent space [13].
- **Spatial UIs in VR (2010s)** – Various VR applications use cylindrical or spherical menus. For example, some game UIs place inventory items in a circle around the player's field of view. The player can spin

around to select items, effectively moving a pointer on a circular manifold. This approach removes arbitrary menu boundaries and taps into natural spatial orientation skills.

- **Infinite Scrolling Carousels** – Common in modern mobile/web design, these UIs allow users to keep swiping through content in one direction without stop. Under the hood, the content index is handled modulo *N*, treating the navigation as movement on a circle (when you pass the last item, you seamlessly continue at the start). This is a simple practical use of a toroidal state space to aid interaction, preventing the cognitive interruption of hitting a hard end.

Conclusion

Continuous attractor models and toroidal topologies, inspired by the brain's navigation circuits, are proving valuable in engineering contexts from autonomous robots to AR/VR and UI design. They offer a way to map states (whether physical pose or abstract UI context) onto a well-behaved continuous manifold that has no edges. In robotics and VR, this yields robust tracking of position and orientation – as seen in RatSLAM and neuromorphic SLAM devices – and even helps AI agents develop an internal "map" for goal finding [12]. In user interfaces, the same principles encourage designing with spatial continuity: arranging interactive states in a continuous space (often implicitly a torus or circle) so that users can navigate fluidly and leverage spatial memory. While the use of a literal grid-cell network for a web or mobile UI remains experimental, the conceptual bridge has been formed – evidenced by proposals to use grid-like codes for conceptual and task spaces [14]. As research and practice continue to intersect, we anticipate more **practical demos** (and perhaps GitHub projects) where a UI's "state space" is treated as a navigable manifold – bringing the elegance of the brain's navigation system to the design of digital navigation and interactions.

Sources:

- Neuroscience of continuous attractors and torus manifolds in grid cells [1] [2]
- Engineering of RatSLAM's toroidal pose-cell network [4] [5]
- Deep learning agent with emergent grid-cell representations [11] [12]
- Neuromorphic SLAM (pose cells and head-direction attractor for AR/VR) [8] [9]
- Grid cells for abstract spaces (conceptual mapping beyond navigation) [14]
- Torus manifold used in pose/view inference model [13]

---

[1] [2] Toroidal topology of population activity in grid cells | Nature
https://www.nature.com/articles/s41586-021-04268-7?error=cookies_not_supported&code=97e3f7f5-4917-4145-a03c-7ce0de3b9f0d

[3] Maintaining and updating accurate internal representations of continuous variables with a handful of neurons | Nature Neuroscience
https://www.nature.com/articles/s41593-024-01766-5?error=cookies_not_supported&code=7d0b0709-f889-42bb-b735-f865c3664d6e

[4] [5] [7] Schematic of the RatSLAM navigation model. The core RatSLAM navigation... | Download Scientific Diagram
https://www.researchgate.net/figure/Schematic-of-the-RatSLAM-navigation-model-The-core-RatSLAM-navigation-model-consists-of_fig1_47812932

[6] Solving Navigational Uncertainty Using Grid Cells on Robots | PLOS Computational Biology
https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000995

[8] [9] NeuroSLAM: A 65-nm 7.25-to-8.79-TOPS/W Mixed-Signal Oscillator-Based SLAM Accelerator for Edge Robotics - Daegu Gyeongbuk Institute of Science and Technology (DGIST)
https://dgist.elsevierpure.com/en/publications/neuroslam-a-65-nm-725-to-879-topsw-mixed-signal-oscillator-based-/

[10] GitHub - google-deepmind/grid-cells: Implementation of the supervised learning experiments in Vector-based navigation using grid-like representations in artificial agents, as published at https://www.nature.com/articles/s41586-018-0102-6
https://github.com/google-deepmind/grid-cells

[11] [12] Vector-based navigation using grid-like representations in artificial agents | Nature
https://www.nature.com/articles/s41586-018-0102-6?error=cookies_not_supported&code=b9888f52-a76e-4e02-80d7-2cafe60036c8

[13] (PDF) Simultaneous Inference of View and Body Pose using Torus Manifolds
https://www.academia.edu/7584843/Simultaneous_Inference_of_View_and_Body_Pose_using_Torus_Manifolds

[14] Spatial Memory | Nolan & Sürmeli Labs
https://nolansurmelilab.github.io/projects/spatial_memory/

[15] Spatial Memory: Why It Matters for UX Design - NN/G
https://www.nngroup.com/articles/spatial-memory/