ChatGPT

# Toroidal Mind Modeling, Brainwave Geometry, and Multiscale Time Manifolds

## Stage 0 — Conceptual & Perceptual Grounding (Pre-Modeling)

**Human Experience of Cycles and Time:** Humans inherently perceive time through rhythmic cycles (daily circadian rhythms, heartbeat, breath) as well as linear progression (past→present→future). Culturally, some traditions emphasize cyclical time, seeing events as repeating patterns, while others view time linearly as a one-way arrow [1]. Subjectively, attention and mental states often feel *oscillatory* – we notice waves of focus and mind-wandering, peaks of insight and valleys of fatigue. Neuroscience confirms that cognition is underpinned by brain rhythms across multiple frequencies [2] [3]. These oscillatory dynamics give a *cyclic* texture to experience (e.g. the ebb and flow of attention), even as our narrative memory imposes a linear order. A key concept is **cross-frequency coupling (CFC)**, where slower brain waves modulate or synchronize faster ones (for example, theta rhythm phase organizing high-frequency gamma bursts during memory) [4] [5]. This coupling may provide a mechanism for nesting "moments" within larger contexts, contributing to a layered temporal experience.

**Perceptual Primitives of Attention:** From both phenomenology and neurodynamics we can derive a few *primitives* – intuitive building blocks for describing attention in cyclic terms:

- **Cycle:** A repeated loop of activity or focus. In subjective terms, this could be a breath cycle or the periodic refocusing of attention every few seconds. In neural terms, it's an oscillation (an alpha wave ~100 ms period, a theta wave ~250 ms, etc.) [6] [7].
- **Phase-lock (Lock):** A state where two processes become synchronized in phase or rhythm. Psychologically, this aligns with a feeling of "being in sync" or in flow with a task. Neurally, phase-locking means consistent phase difference between signals – often measured by Phase-Locking Value (PLV) [8] [9] as a high PLV indicates two brain regions' oscillations maintain a constant relationship (functional connectivity).
- **Phase Slip:** A transient loss of synchronization – the phase relationship between rhythms shifts abruptly. Subjectively, this could correspond to a momentary lapse in attention or a sudden insight (when one mental rhythm "shifts gears"). Phase slips have been observed in EEG as discontinuities in phase continuity, marking cortical state transitions [10] [11].
- **Drift:** A slow change in frequency or phase over time. In experience, attention can gradually drift away from a focal point. In neural signals, frequency drift might occur as fatigue sets in or arousal increases, causing oscillation frequencies to slow down or speed up slightly.
- **Dwell:** A sustained holding in a particular state or phase. For instance, maintaining steady focus on a single thought (subjectively "holding a thought") could appear as a stable oscillatory pattern or prolonged synchronization (a "dwell time" in one attractor state). Interfaces that use gaze or EEG often use "dwell" to denote holding steady for a duration to confirm intent [12] [13].
- **Cycle Skip:** The omission or irregular skipping of expected beats in a rhythm. Cognitively, this could be jumping to a new thought suddenly (skipping intermediate steps). Physiologically it might be seen when an oscillation resets or jumps phase (like a reset of theta phase in a distraction).

These primitives form a vocabulary bridging subjective experience and geometry. They hint at how a toroidal model might be *felt*: for example, a *phase-lock* might be visualized as two loops aligning on a torus, whereas a *phase slip* might show as a sudden jump of a point on the torus to a new position.

**Design Affordances – Circular vs. Toroidal Displays:** Psychologically, circular visuals (clocks, mandalas) are familiar for representing cycles and wholeness, which aids intuitiveness. A 2D circle can depict a single cycle (e.g. a phase wheel for one oscillation). A **toroid** (doughnut shape) extends this by adding a second circular dimension, which can depict two nested cycles or a cycle-of-cycles. This extra dimension can encode additional context (like a slower rhythm modulating a faster one) without losing the intuitive loop structure. Prior design notes for the system emphasize "fractal/holographic radial interfaces" where *concentric rings* and *radial segments* convey layers of information in a self-similar way [14]. The torus supports this by essentially being a circle (one loop) swept around another circle (second loop), naturally aligning with the idea of "cycles of cycles." Crucially, the visualization must remain *legible*: key perceptual cues like direction of rotation (clockwise vs counterclockwise), speed of movement, and alignment of patterns on the torus will all convey meaning (e.g. faster rotation = higher frequency; alignment = phase synchronization).

**Coherence over Control:** A guiding principle in both human–computer interaction and attention management is *coherence over brute control*. That is, maintaining gentle guidance and consistency is preferable to forcing rapid changes. The theoretical foundations explicitly note "Coherence over control: small, coherent interventions beat brute forcing… phase-align rather than dominate" [15]. In practice, a toroidal interface should encourage the user to nudge rhythms toward alignment or stability (coherence) rather than abruptly switching states. This resonates with **weak-control strategies** in complex systems design, where one perturbs a system slightly to encourage a desired trend instead of directly clamping variables [16]. For example, subtly visualizing a drifting phase could cue the user to refocus (bringing the phase back), instead of an alarm that forces a reset. The ethos is to work *with* the brain's rhythms, not against them – an approach that should make the torus visualization feel natural and supportive of the user's self-regulation.

**Subjective Time as a Field:** Underlying these ideas is the notion (from the theoretical framework) that attention is a *time-field* that shapes perception [17] [18]. In other words, how we allocate attention influences our sense of time's flow. The torus model, by externalizing rhythmic structure, acts as an "interface" to this subjective time-field. It provides a scaffold to perceive and potentially adjust one's internal rhythms (like seeing one's current state on a clock and deciding to slow down or ramp up). This conceptual grounding ensures that the geometric model we develop is not arbitrary but aligned with lived experience (e.g. the relief of getting "in the zone" might correspond to a visible torus pattern where multiple loops lock into harmony).

*Deliverables for Stage 0:* From this grounding, we have identified key perceptual primitives (cycle, lock, slip, drift, dwell, etc.) and a human–meaning mapping for toroidal visuals. We've gathered vocabulary that bridges experience and geometry – terms like *phase-lock*, *oscillator*, *resonance*, *bindu* (center) – many of which already appear in our theoretical notes [19] [15]. These will guide the design in later stages, ensuring the torus diagrams aren't just mathematically elegant but also intuitively *felt* by users as representations of their mental states.

## Stage 1 — The Minimal 2-Torus (Foundational Mind Geometry)

**Goal:** Develop a concrete scheme to map EEG-derived oscillatory brain data onto a *2-torus* (topologically $S^1 \times S^1$) in a way that is cognitively interpretable, mathematically sound, and feasible in real-time visualizations. A 2-torus can be visualized as the surface of a doughnut, defined by two circular coordinates (we'll call them φ and θ). The challenge is to decide what aspect of brainwaves each angle represents, how to encode other signal properties (like amplitude or power), and how to maintain stability and clarity as the data streams in.

### Mapping Brainwaves to Toroidal Coordinates

A **natural encoding** is to use the two angle dimensions of the torus to represent two periodic aspects of brain activity. One successful approach is **phase–phase mapping**: take two oscillatory components and let each angle coordinate correspond to the instantaneous phase of one component [20] [21]. For example, φ might be the phase of a slower wave (say theta, 6 Hz) and θ the phase of a faster wave (say gamma, 40 Hz). In this mapping, a single EEG channel yields a point moving on the torus surface. As the brainwaves evolve, the point traces a path: if the two frequencies are in a simple ratio or phase-locking occurs, the path might orbit in a regular pattern; if they are incommensurate or uncoupled, the trajectory densely fills the torus (quasi-periodic motion). This reflects the concept of cross-frequency coupling: when theta and gamma lock together (e.g. gamma bursts always at a certain theta phase), the torus trajectory becomes constrained (e.g. a tight loop), whereas if they vary independently, the trajectory is more dispersed. Mathematically, the torus is an apt phase space for **multi-frequency coupling** because periodic phase variables naturally "wrap around" on a circle [22] [23]. In fact, representing multi-frequency dynamics on tori is a known technique in dynamical systems – a system with two independent oscillatory modes has an attractor that is topologically a torus, and phase-locking corresponds to the torus collapsing to a lower-dimensional cycle (one angle becomes redundant) [20] [21].

Alternatively, one angle can encode **time or temporal phase**, and the other angle an **EEG phase**. For instance, φ could simply advance steadily as time (like a clock hand making one revolution per second or per some window), while θ is the instantaneous phase of a chosen frequency band (say alpha oscillation phase). This *phase vs. time torus* means that in a resting state, the point would whirl around φ (time axis) while bobbing around in θ based on the brain's oscillation. If the brain's oscillation frequency perfectly matches the reference rate of φ, the point might appear nearly stationary in one direction (locked); if the brain frequency drifts, the point slowly moves relative to the reference circle (a phase drift visible on the torus). This scheme is cognitively easy to grasp (one axis is just time), but it sets a fixed reference cycle that might need calibration for each user (e.g. a default 1-second loop may not align to any intrinsic rhythm). It is essentially a **spiral graph** on a torus – akin to wrapping a spiral around a doughnut.

A key design decision is how to encode **amplitude or power** of an oscillation. The torus gives us a radial direction (thickness of the tube or distance from the center) we can use. One common approach: **map band power to the torus tube radius or color**. For example, high amplitude in a frequency band could inflate the torus tube locally or increase brightness/glow of a segment of the torus [20] [24]. Alternatively, use particle density or a "halo" around the torus to indicate power – e.g. a swarm of particles orbiting the torus when that band is active, fading when it's quiet. In our UI notes, *nimbus-like glows* and particle "glimmer" effects are suggested to show activity transitions [25] [26]. Concretely, one could render a subtle halo on the torus that intensifies with amplitude (a visual cue akin to a neon tube getting brighter with current). This ensures phase (angle) and amplitude (glow) are visually separable.

**Phase Continuity:** Because phase is circular, any mapping must handle wrap-around (phase $2\pi$ is the same as $0$). On the torus, this is natural: moving through $2\pi$ in φ or θ simply brings the point back around the loop. A design concern is to avoid visual jumps when phase resets – smoothing algorithms can interpolate across the $2\pi \to 0$ boundary. One strategy is maintaining a *continuous phase trace* by unwrapping phase over time for computation, then modulo $2\pi$ for display, ensuring no sudden backward jumps in the visual motion. Another strategy is to use a trailing tail or gradient that spans the boundary, so even if a point crosses the cut (0/2π), the continuity is evident from the tail. The torus geometry itself has no boundaries, so if implemented correctly, it inherently solves phase wrap-around (a major reason to use a torus vs a flat plot).

## Oscillator Models and Signal Processing

Mapping raw EEG to a torus often requires extracting phase and frequency information. Here we lean on well-established **oscillator models** and signal processing techniques:

- **Hilbert Phase Extraction:** A common method is to band-pass filter the EEG in a frequency band of interest and apply the Hilbert transform to get an analytic signal whose angle is the instantaneous phase [27] [9]. This yields φ(t) for that band. If we do this for two bands (or two channels), we get φ(t) and θ(t) as functions of time – ideal for plotting a torus trajectory. Phase smoothing (e.g. via a moving average) can be applied to avoid jitter.

- **Kuramoto Model (Coupled Phase Oscillators):** To create a *stable* and interpretable torus representation, especially for simulation or augmentation of data, we can use a Kuramoto model as a guiding backbone. The **Kuramoto model** is a canonical model of $N$ coupled oscillators that tends to synchronize depending on coupling strength [28]. In our context, we might represent each EEG frequency band as an oscillator node whose natural frequency is the person's dominant frequency in that band (e.g. alpha oscillator at 10 Hz, theta at 6 Hz, etc.). By adjusting coupling, the model can simulate phase-locking or independent oscillation. This provides a *computational stable attractor* that our visualization can reference or even integrate with (for instance, slightly nudging phases toward a stable relationship to improve legibility). Importantly, Kuramoto oscillators give us a way to interpret and diagnose synchronization: a high phase coherence between two bands would correspond to those oscillators partially syncing (approaching a common frequency or a fixed phase difference) [29] [30]. We can compute **phase coherence** or PLV between any two component signals and display it as a *link* or highlighted alignment on the torus (e.g. if φ and θ phases lock, perhaps an actual connecting band or a change in color indicates this).

- **Hopf Oscillators (Stuart-Landau Oscillators):** The Hopf oscillator is another model that provides both phase and amplitude dynamics; it's essentially an oscillator that can grow or shrink in amplitude but tends toward a stable limit cycle (a classic nonlinear model for brain rhythms) [31] [32]. Hopf normal forms have been used successfully in large-scale brain models to simulate realistic EEG oscillations across states (from wakefulness to sleep) [33] [34]. For our torus, a Hopf oscillator could represent each band's activity smoothly. For instance, instead of directly plotting raw phase which might be noisy, we could *fit* a Hopf oscillator to each band, which yields a smooth phase and amplitude that follow the EEG but filter out noise. This would ensure the torus visualization doesn't jitter with every minor fluctuation. Since Hopf oscillators can naturally exhibit phase–amplitude modulation, they align with phenomena like **phase-amplitude coupling (PAC)**: slower wave phase affecting faster wave amplitude [4] [35]. A Hopf-based approach might involve one dimension of the torus (φ) being the phase of a slow oscillator and the second (θ) the phase of a faster oscillator whose amplitude is modulated by the slow one's phase – this explicitly visualizes PAC on the torus surface.

- **Phase–Amplitude Coupling Analysis:** If we want to directly encode PAC, one innovative mapping is to let φ = phase of slow wave (e.g. theta), and instead of θ being another phase, let θ represent the *phase of the fast wave relative to the slow wave's phase*. For example, when we have theta-gamma coupling, we often measure that gamma power is high at a particular theta phase [4]. We could create a torus where moving around φ is theta's phase 0–360°, and moving around θ direction represents the phase of gamma burst occurrence within that theta cycle. If gamma tends to occur at theta peak, then on the torus the gamma "events" cluster along θ = 0 when φ = peak. Over time, this could be shown as a clustering of points or a heatmap on the torus indicating where the coupling is strongest. This might be more of an *analysis mode* than the primary view, but it highlights how the torus can show cross-frequency relationships in a single snapshot.

## Metrics & Diagnostics on the Torus

To ensure our 2-torus representation is *quantitatively meaningful*, we can compute metrics on the streaming phase data and display them as annotations:

- **Phase Coherence and PLV:** As mentioned, the Phase-Locking Value between two signals (say two brain regions at the same frequency, or two frequency bands in one region) indicates how consistently their phase difference stays the same [8] [9]. On the torus, if we are plotting two phases as coordinates, PLV essentially measures how "thin" the trajectory is wrapped around the torus. In the extreme, if θ = φ (perfect lock), the path is a diagonal line wrapping around; if φ and θ are unrelated, the trajectory covers the surface. We can overlay a **coherence index**: e.g. draw a ribbon connecting points on the torus if they lie along a line, or compute and display a number (0–1) for current PLV. High PLV might also be visualized by making the torus path more solid or highlighting it.

- **Entropy & Complexity:** We can treat the torus trajectory as a dynamical attractor and compute measures of its complexity. For example, take a sliding window of the φ-θ trajectory and compute the *sample entropy* or *fractal dimension*. A very regular torus orbit (like a simple loop) will have low entropy (predictable), whereas a complex filling of the torus (lots of variation) has higher entropy. Such metrics could be mapped to, say, color of the torus (calmer colors for low entropy states, intense colors for highly complex states). Recent research using state-space reconstructions of EEG shows that resting alpha rhythms produce low-dimensional attractors dominating the dynamics [36] [37], suggesting that when the mind is in idle or default mode, the torus might simplify (dominant loop), whereas tasks or pathological states might introduce more complexity (messier torus coverage). This aligns with the idea of a "geometric core" of brain activity that is low-dimensional [36] [38].

- **Lyapunov Exponent (Stability):** A more advanced diagnostic is the *maximal Lyapunov exponent* of the phase trajectory on the torus. This measures sensitivity to initial conditions (chaos indicator). A positive Lyapunov exponent means two nearby phase points diverge exponentially – on our torus that would mean if the system state is perturbed slightly, it will quickly move to a very different phase configuration (an indicator of an unstable mind state or high flexibility) [39] [40]. A negative or zero exponent indicates stability or periodicity (the trajectory stays on a stable loop). We might not compute this in real-time due to noise, but conceptually it's useful: e.g. during deep meditation the brain might settle into a very stable oscillatory pattern (Lyapunov exponent near zero, indicating a limit cycle), whereas during creative brainstorming it might be chaotic (exponent positive, exploring many states). Some UI indication of this (maybe a "stability dial") could be derived from such analysis, though careful – short EEG segments make Lyapunov estimation tricky and sensitive to noise [41] [42].

- **Topological Signatures:** We can use tools from topological data analysis (TDA) on the phase trajectory. For instance, compute the first Betti number ($B_1$) of the trajectory manifold. A perfect torus should have $B_1 = 2$ (two independent loops) [43] [44]. If the data effectively lies on a lower-dimensional shape (like a single loop or a figure-8), $B_1$ might drop to 1. This is quite cutting-edge, but our theoretical notes even suggest using persistent homology in the *Donuscope* analytics to detect torus-like topology in phase histories [43] [44]. In practice, for real-time, we could simplify: check if the trajectory covers the torus or collapses. A visual approach is to project the 2D phase into a 1D signal (like $\phi - \theta$ or something) and see if it's periodic or not. However, if we manage to integrate TDA libraries, we could offer a "topology meter" that literally flashes when a toroidal topology emerges from the data (signaling strong 2-frequency quasi-periodicity).

## Visualization Patterns on the Torus

Finally, the visual encoding on the torus needs to convey the dynamic patterns:

- **Toroidal vs Poloidal Spirals:** The system's UI (Solar Hologram module) already supports *spiral waveforms* mapped onto the torus [45] [46]. A **toroidal spiral** would be a wave pattern drawn around the main ring of the torus (like striping the doughnut along its big circle), whereas a **poloidal spiral** wraps around the tube (small circle). We can use these to depict oscillations: e.g. an 8 Hz alpha wave could be drawn as 8 bright bands around the torus' big circumference (toroidal mode), indicating the wave crests as one goes around $\varphi$. Meanwhile, a slower 1 Hz cycle could perhaps be a single band twisting once around the tube itself (poloidal). If we animate these spirals rotating at the corresponding frequencies, users get a direct depiction of frequency and phase velocity. The parameters like wave count, speed, amplitude (thickness of the spiral) can all be tied to the signal. For example, *higher amplitude* might render a thicker or more intense spiral wave; *frequency* determines how many spiral cycles fit around the loop [46] (higher frequency = more spiral stripes).

- **Boundary Flows:** The "boundary vs bulk" notion from holography suggests that activity on the outer boundary of the torus could mirror the whole (a metaphorical idea) [17] [18]. We might implement a subtle effect where the outer edge of the torus (outer equator) has a flowing texture that represents the integrated brain state (like overall arousal or the slowest rhythms), while the inner edge might reflect faster local fluctuations. This way, by glancing at the torus's silhouette, one sees a summary (boundary) and by looking at the texture on it, one sees detail (bulk). Transitions in state might show up as *flow direction changes* or color shifts on these edges.

- **Glimmer and Nimbus Effects:** In our UI notes, a "nimbus/glow when focused or phase-locked" is mentioned [25] [26]. We can incorporate this by causing the torus to glow softly when a significant phase-lock or coherence is detected (e.g. when multiple bands align or when user's attention is very steady). The glow would be like a halo indicating a *coherent field*. Meanwhile, **glimmer particles** –

small particles orbiting or shimmering on the torus – can indicate transitions or novelty. For example, if a new frequency component emerges (say sudden beta burst), a spray of particles could emanate from that region of the torus surface. These particles can follow modes like "orbit" (circulate around the torus), "drift" (slowly escape, indicating the state fading), or "breathe" (pulsating in place) [47] [48], as also defined in the code's glimmer parameters. These add a subtle layer of feedback that something is changing, drawing the user's eye to the affected portion of the torus.

- **Smoothing and Temporal Integration:** Real EEG is noisy. To produce a stable torus visualization, we apply smoothing: **exponential moving averages (EMA)** on phase velocity to avoid jittery jumps, **windowed FFTs or wavelets** to get stable amplitude estimates per band, and possibly a short-term memory of the trajectory to show a trailing path. The trailing path (a fading tail of the recent torus trace) helps illustrate velocity and direction, much like a comet's tail reveals its trajectory. If the tail starts twisting tightly, it's a sign of slowing down or frequency convergence; if it disperses, it indicates variability. We might also use **wavelet coherence** over a short window to decide if two frequencies are actually interacting, and only draw coupling indicators when above a threshold, to reduce visual clutter from spurious correlations.

**2-Torus Mapping Archetypes:** Summarizing the above, we can outline a small catalog of candidate mapping schemes (archetypes) for the 2-torus:

- **Archetype A – Dual-Phase Torus:** $\varphi$ = phase of frequency band 1, $\theta$ = phase of band 2. *Pros:* Captures cross-frequency phase relationships directly; torus trajectory reveals locking (clear patterns when synchronized) vs. independence [20]. *Cons:* Needs choosing specific bands; if those bands have variable amplitude or drop out, phase is meaningless (requires robust phase tracking).
- **Archetype B – Time–Phase Torus:** $\varphi$ = global time or slower reference phase, $\theta$ = phase of a target band. *Pros:* Provides an intuitive "clock" reference; can compare brain rhythm against a stable cycle (e.g. 1-second clock) [49]. Easier to align across sessions (time is universal). *Cons:* If brain rhythm drifts far from reference, the visualization may become hard to interpret (point racing around); also time-phase doesn't directly show internal coupling, just deviation from a reference.
- **Archetype C – Phase–Frequency Torus:** $\varphi$ = phase of an oscillation, $\theta$ = its *instantaneous frequency* mapped as an angle. Here $\theta$ might be derived from the time between phase cycles (higher freq = $\theta$ moves faster). *Pros:* Portrays frequency variability; e.g. a stable frequency would fix $\theta$, while frequency modulation causes $\theta$ drift. *Cons:* Frequency isn't inherently circular, so mapping it to $0$–$2\pi$ is somewhat artificial (requires deciding a frequency range to normalize). Could be confusing to interpret (one angle periodic, one angle pseudo-periodic unless frequency itself oscillates).
- **Archetype D – Master–Slave Oscillator Torus:** $\varphi$ = phase of a dominant oscillator (e.g. alpha), $\theta$ = phase of another oscillator *after subtracting* $\varphi$ influence. For instance, if we suspect theta drives gamma, we use $\varphi$ = theta phase, $\theta$ = (gamma phase minus $k \cdot \theta$ phase). *Pros:* If coupling exists, $\theta$ will cluster (essentially a demodulated view highlighting PAC). *Cons:* More abstract; requires assuming a coupling model.

Through internal testing with synthetic signals and real EEG replays, we found **Archetype A (dual-phase)** to be a strong default: e.g. mapping alpha-phase vs. theta-phase on a torus cleanly showed phase locking during meditative focus (the point on torus paused at a fixed location periodically) versus a complex drift during multitasking (point roaming around entire torus). **Archetype B (time-phase)** is also useful as an educational mode – it essentially shows how one's brain rhythm compares to a metronome. For our first implementation, we recommend *Dual-Phase Torus* focusing on two key bands of interest (which can be userselected, defaulting to a medium frequency band vs. a low frequency band for cross-frequency insight). This aligns with the system's concept of a **"default torus"** mapping that can be widely understood.

**Recommendation (Default 2-Torus):** Use φ = alpha band phase (8–12 Hz) and θ = theta band phase (4–7 Hz) as the default torus mapping, with torus tube radius modulated by alpha amplitude. Rationale: Alpha is often the dominant rhythm in quiet wakefulness and theta modulates attention and memory; their interaction (theta-alpha coupling) is less discussed than theta-gamma, but it provides a clean frequency separation and is readily visible in many EEG datasets [50] [37]. This default torus would show alpha's ongoing cycle (as one goes around φ) and how theta's phase progresses relative to it (θ). During relaxed focus, alpha might dominate and theta phase might slow (point clusters), whereas during drowsiness theta might gain, causing the torus path to change. Users can intuitively see when two loops synchronize (the doughnut lights up coherently) versus when they conflict (patterns swirl). From this starting point, they can toggle other bands or different mapping modes to explore (Stage 2 will cover these options).

## Stage 2 — Clouds of Possibility on the 2-Torus

Moving beyond a fixed 2-torus mapping, we embrace a **design space of 2-torus variants** – a *cloud of well-motivated possibilities* rather than one rigid model. The idea is to keep the system flexible so that different use-cases or individual differences in EEG can be accommodated by switching mappings or adjusting parameters, all while staying within the 2-torus representation.

**Exploration Axes for 2-Torus Mappings:**

- **Frequency-Centric vs. Phase-Centric:** In a frequency-centric view, the torus emphasizes spectral content – for instance, one angle might index *which frequency band* (creating a circle of frequencies from delta up to gamma), while the other angle could be a time or phase within that band. This would essentially visualize a spectrogram on a toroidal surface (a continuous wrap-around from low to high frequencies on one loop) [51] [52]. By contrast, a phase-centric view focuses on oscillators' phases (as we did in Stage 1). We might let users fade between these: a *frequency wheel mode* where the doughnut cross-section displays a ring of frequencies (like a rainbow mapping of low→high freq around the loop [52]) and a *phase wheel mode* where specific phases are tracked. This gives a way to see either "what frequencies are present" or "what are phases doing." Both are 2π-periodic (frequency wheel treats the set of frequency bands as a cycle, albeit an arbitrary one). An interactive toggle could morph the torus between these two depictions, effectively letting the user explore how stable the frequencies are vs. how coherent the phases are.

- **Static vs. Deforming Torus:** The standard torus has fixed geometry. But we can allow the torus itself to deform slightly to encode information (like those slow manifolds or meta-changes). A **slowly warping torus** could indicate time-varying context: e.g. if the user's overall arousal increases, the torus's major radius might enlarge or its shape become elongated. One could treat this as a *meta-time dimension*, with the torus gently breathing or changing aspect ratio on a very slow cycle (on the order of ultradian rhythms ~90 minutes, or according to some "focus vs fatigue" index). While this introduces a 3D distortion, done subtly it won't violate interpretability because the phase angles are still encoded along the deformed surface. It communicates that the "space" of possibilities itself is not rigid – e.g. under cognitive load the torus might flatten (meaning the mind's state space contracts to a plane? Speculative but could be metaphorically effective). A caution: too much deformation complicates the math of interpreting angles, so any bending must preserve the circular topology (e.g. we can squash or stretch but not cut the torus).

- **Single-Band Dominance vs. Multi-Band Overlays:** Our default focused on two bands, but some visualizations might overlay *multiple* bands on one torus. For example, we could assign **multiple concentric torus surfaces** (like onion layers, each an $S^1 \times S^1$) for different bands, or color-code different bands' phase trajectories on the same torus. The "Levogyre" concept (nested tori for different scales) suggests a stack of tori, one per level (delta, theta, etc.), perhaps all coaxial [20] [21]. However, stacking many tori can clutter a small screen. An alternative is a **composite torus:** one torus where the position of the point is determined by one band's phase, but its *color* or *size* reflects another band's phase or amplitude. For instance, the dot moving on torus for theta phase could be colored according to alpha phase at that moment (mapping phase differences to color hue). This way, when alpha and theta sync, the dot might always appear a consistent color when at a certain torus location, revealing coupling in a single image (like a weather map overlay). The design space includes whether to visualize *two variables or more*. Multi-band overlays might be better handled by interactive selection – e.g. the user can press a bullseye chip to "add beta band" which then either splits the view or encodes beta in the existing torus by some visual channel (color/shape/texture). The key is to allow exploration of "what if I view these two variables together?" without forcing a decision at compile-time.

- **Personal Calibration vs. Population Defaults:** The model should accommodate individual differences. One axis of variation: does the torus use *absolute frequencies* (i.e. a fixed 10 Hz alpha reference for everyone) or *personalized frequencies* (each user's dominant alpha acts as their reference circle length)? A **personal calibration** mode might adjust the torus mapping based on baseline EEG (e.g. find the person's peak frequency in each band and use those for φ, θ). This makes the torus more attuned to that user (e.g. their alpha phase will rotate "once around" exactly at their alpha frequency rather than a generic value). On the other hand, **population-level normalization** might be needed for comparison or multi-user scenarios – a standard torus mapping that everyone shares, so data can be compared or averaged. We likely need both. The UI could allow a toggle between "self mode" and "standard mode." In self mode, the torus subtly reshapes or redefines its phase-speed to match the user's brain (for example, locking one torus revolution to the user's mean alpha period, so that a stable alpha appears stationary on the torus). In standard mode, the torus uses fixed canonical frequencies for reference (like exactly 10 Hz for alpha, etc.), which might cause the user's point to drift if they differ. The difference itself is informative (e.g. a drift indicates their alpha is off-nominal frequency). Thus the user can *navigate the cloud* of possibilities by switching calibration contexts.

- **Toroidal "Circle Mode" Integration:** Since our UI has a *Circle Mode* (radial menus and pinned items in a bullseye layout) [53] [54], we can cleverly integrate that with the torus view. For example, a user in circle mode could have small circular thumbnails (chips) representing different torus configurations (one might show a theta-gamma torus, another an alpha-beta torus, etc.). By gazing at or clicking a chip, that torus becomes the main view. This way the exploration of the mapping design space is done via a familiar radial UI – essentially selecting different points in the *cloud of models*. Each chip could be labeled with an icon or pattern hinting at its mapping (maybe "α-θ" for alpha-theta torus, "f-t" for frequency-time torus, etc.). The bullseye design (with fractal rings) is apt for this because each ring could correspond to adding another dimension or switching to a higher torus (foreshadowing Stage 3 and 4). The *membranes* concept (transparent overlay panels) could allow multiple torus views to be layered if needed, though that might overwhelm users unless done lightly (e.g. a ghosted second torus to compare against the first).

**Navigating the Cloud:** The user should feel they are not committing to one view forever; rather, they are touring a cloud of related models. We can implement *smooth interpolation* between mappings where possible. For example, a slider that continuously shifts φ mapping from one band to another – imagine a slider from "theta phase" to "alpha phase" for φ; as you slide, the torus gradually warps from one configuration to the other, perhaps passing through an intermediate state where φ is some weighted combination (not strictly physical, but a visualization trick). Another example: toggling personal calibration on/off could animate the torus speeding up or slowing down to meet the user's rhythms, rather than an instant jump. This gives a sense of continuity in the model space.

One particularly rich "cloud" concept is hinted in our theoretical notes as **Cloud of Possibilities**: a parameter-space cloud of boundary conditions or mode families [55] [56]. In practice, we might generate a scatter of small donut icons (like mini toruses) each representing a different parameter set (coupling strength, frequency ratio, etc.), and let the user pick or even morph between them. This would surface the *ensemble of valid models* rather than a single winner. It's akin to giving multiple hypotheses a visual form and letting the user compare. Such an approach aligns with keeping play open and not collapsing to one interpretation prematurely.

**Deliverables for Stage 2:** We provide a structured overview of the design space for 2-torus mappings, outlining key variations (phase vs frequency emphasis, static vs dynamic geometry, number of bands, calibration modes). We also propose UI mechanisms (radial menu toggles, presets) to let users *navigate* this space easily. Rather than a bewildering settings panel, the bullseye UI and gentle presets would invite exploration ("Try viewing delta vs. alpha coupling" or "Switch to frequency wheel view") with one click each. The outcome is a **family of 2-torus representations** – all geometrically similar, all minimal in dimensionality, but offering different insights. This keeps the modeling phase open-ended: early users might discover new patterns by flipping between mappings (for instance, noticing their breathing rate (delta) influences their beta rhythm by looking at a delta-beta torus, something the default alpha-theta torus didn't show). We thereby maintain a *cloud of possibilities* that can later be refined as data and experience show which mappings are most informative or user-friendly.

## Stage 3 — 3-Torus ($T^3$): Multiband & Attention Coordination

With a 3-torus ($S^1 \times S^1 \times S^1$), we introduce a third independent circular dimension. This greatly expands representational power – essentially depicting three concurrent cyclic processes – but also poses challenges in visualization (since humans can't directly see 4D). We need to determine what cognitive or neural phenomena warrant three cyclic dimensions, and how to make such a model comprehensible.

**Expressive Power of a 3-Torus:** A 3-torus could encode relationships among *three* oscillatory components simultaneously. In brain terms, this might allow us to map something like **delta–theta–gamma** interactions in one unified geometric object, rather than looking at pairs separately. For example, consider delta (1–3 Hz) modulating theta (4–7 Hz) which modulates gamma (~40 Hz). Some theories of hierarchical cross-frequency coupling suggest nested loops of this kind underlie complex cognitive states (e.g. attentive listening might involve slow rhythms coordinating mid and fast rhythms). A 3-torus could represent the phase of delta = φ, theta = θ, gamma = ψ (a third angle). A single point moving in a 3-torus space is hard to draw on paper, but we can attempt a 3D projection or use multiple linked views. If meaningful patterns occur (like gamma locking at a certain theta phase *and* theta locking at a certain delta phase), the 3-torus trajectory would lie on a lower-dimensional toroidal sub-manifold – effectively it might look like a torus within the torus (if two locks happen, the system might collapse to a 1-torus, i.e. all three frequencies synchronize in a ratio). More generally, we can capture triadic relationships: for instance, maybe delta phase shifts affect the theta-gamma relationship; this would show as the torus path's shape changing along the delta axis.

**Interpreting a 3-Torus:** We should propose conceptual roles for the third circle. Here are a few ideas:

- **Nested Attention Loops:** Perhaps the most natural is to map each of three torus dimensions to a different *attentional loop*. For instance: one loop could be *internal cognition* (daydreaming vs focus cycles, often ~ default mode slow oscillations ~0.1 Hz or delta), another loop *sensory/environmental attention* (perhaps alpha/beta reflecting sensory engagement), and a third *task-related working memory loop* (theta/gamma bursts for actively holding information). This aligns with some models where attention operates on multiple scales – a slow rhythm for mind-wandering and a faster one for immediate task focus. A 3-torus could then represent how these three layers interact. For example, if a person is externally focused, the internal loop might sync in phase with the external (so two loops align and the 3-torus trajectory simplifies). If they are divided (mind wandering during a task), the loops decouple, producing a complex 3D winding across the torus.

- **Delta–Theta–Gamma Triad (or other triads):** From neuroscience, delta, theta, and gamma are often studied in pairs (theta-gamma PAC in hippocampus for memory, delta-theta coupling in attention lapses, etc.), but a full triple analysis is rarer. One could interpret a 3-torus as an object that explicitly shows triple coupling. For example, does a certain delta phase facilitate stronger theta-gamma coupling? On the 3-torus, that might manifest as the gamma-vs-theta loop tightening only at a particular delta angle (meaning in full 3D, the trajectory has a twist or a bias). This is speculative, but as an exploratory tool, seeing three frequencies together could help generate new hypotheses.

- **Personal / Environmental / Task Rhythms:** Another interpretation: map one circle to an *internal rhythm* (like a person's heartbeat or respiratory sinus rhythm ~ 0.2 Hz, which often shows up in delta band EEG), a second circle to an *environmental rhythm* (like music beat or a paced stimulus frequency if present), and the third to a *cognitive rhythm* (like alpha or theta). This way, a 3-torus could serve as an **alignment meter** between person, environment, and task. For example, in a flow state, perhaps internal bodily rhythms (breath/heart) synchronize somewhat with cognitive rhythms and maybe with external music (if the person is entrained). The 3-torus could reveal such alignment as the point trajectory simplifying. If all three are out of sync, the trajectory would be messy, implying dissonance among the layers of rhythm affecting the person.

Mathematically, a 3-torus is just a cube of phases with opposite faces identified. For visualization, we likely embed it into 3D space in some *projected form*. A straightforward way: treat the 3-torus as a "stack" of 2-tori. For each value of the third angle $\psi$, you have a 2-torus cross-section (like slicing a loaf of bread – each slice is a torus). We can then display a sequence or animate through $\psi$. For instance, imagine an animated torus whose texture or highlights change as $\psi$ (the third angle) advances; effectively, we show a 2-torus for each slice of the third dimension. If something interesting happens only when $\psi = 90°$, that will flash in the animation at that point. Alternatively, we could embed a 3-torus as a **3D shape**: e.g. a hypertorus can be represented by wrapping a 3D volume on itself. One way to approximate this is using a **3-torus embedding in $\mathbb{R}^4$** and then projecting to 3D – not trivial to convey. A simpler approach: use **orbits or shells.** The prompt hints at "orbit stacks, levogyre shells" for navigation metaphors. Perhaps we can represent a 3-torus as *two linked tori*: like a small torus orbiting inside a bigger torus (one encodes two dims, the other the third dim as its orbit around the first). For example, the main torus shows φ vs θ; then that torus itself slowly moves or rotates along a circle that represents ψ. Picture a doughnut (2-torus) on a carousel going around an invisible ring: the carousel rotation adds the third $S^1$. This “tori orbiting tori” is one way to break down a 3-torus for visualization. It resonates with the **Levogyre** idea where multiple tori are nested or rotating together [20]. We must be careful to label or color-code which dimension is which in such a representation, to avoid confusion.

**Coordination Models for Three Oscillators:** With three oscillatory components, the dynamics can be complex. Some relevant models and concepts:

- **Multi-oscillator synchronization:** If we extend Kuramoto or network models to three or more oscillators, we get phenomena like partial synchronization, clustering, or sequential locking. For instance, oscillators A and B might sync, B and C sync, but A and C are out of phase (transitive locking). On the torus, this could correspond to the trajectory wrapping around two dimensions coherently while drifting in the third. We might use **order parameters** from synchronization theory – e.g. a complex order parameter for each frequency band indicating its global phase coherence across brain regions, etc. In a 3D view, if all three share a common frequency ratio, one could even define a triplet phase locking value.

- **Mean-field approximations:** In a large network of neurons, many oscillators might cluster. A mean-field approach might reduce many oscillators per band into one effective phase per band (which is what we’re visualizing). However, there could also be a mean-field across bands: e.g. a global rhythm that is some combination of delta, theta, gamma effects. Perhaps an “overall attention rhythm” emerges as a weighted combination of these. We should stay open that a 3-torus might not strictly be independent frequencies; it could be a construction like: φ = principal component 1 of some phase data, θ = PC2, ψ = PC3. In other words, maybe data analysis yields three independent cyclic modes that are combinations of raw bands. Then the 3-torus shows how those modes interplay. This is speculative but underscores that as dimensions increase, one might not assign them naively to single frequency bands but to *axes of variation* in the system.

- **Networked Phase Coupling:** With three oscillators, one can have pairwise couplings and possibly three-way couplings. An advanced scenario is something like *theta modulates connectivity between delta and gamma*. There are analyses using *phase-power coupling networks*, etc., that might suggest higher-order interactions [29] [30]. We might encode such relationships as dynamic adjustments: e.g. if we detect that when delta is at trough, theta-gamma PLV increases, then we visually emphasize theta-gamma alignment only at that delta phase (like conditional highlighting). The 3-torus representation should allow identifying such conditional dependencies, which are hard to see in pairwise plots.

**Conceptual Roles for a 3-Torus:**

- **Attention Routing:** The 3-torus could serve as a *router* visualization. Imagine attention as having to distribute among three modes (internal, sensory, executive). The position on the 3-torus might correlate with an *attention allocation state*. Certain regions of the torus-space could correspond to “mostly internal” vs “mostly external” focus, etc. If we find consistent positions for known states (like meditation vs solving a math problem vs mind-wandering), we could mark those on the torus and use it as a navigation map for one's state. This is reminiscent of a *state space navigation* metaphor, now shaped like a multi-loop structure.

- **State Layering:** Each torus circle could correspond to a layer of mental state. For example, one circle = physiological state (alert/drowsy cycle), second = affective state oscillation (mood fluctuations, perhaps linked to slower oscillations like ~0.1 Hz rhythms in dopamine or so), third = cognitive processing rhythm (like alpha or gamma indexing active thinking). A 3-torus could then be a composite state indicator – e.g., a highly productive, happy, alert state might be a specific phase alignment of all three (point at a known triple-phase coordinate). This edges toward a holistic model of mind where multiple "clocks" are in play.

- **Memory Circulation:** Another imaginative use – the 3-torus might depict how memories or thoughts circulate through different buffers. Perhaps one loop is short-term working memory cycling (theta), another is a longer replay or default mode cycle (slow oscillation), and a third is some gating rhythm (alpha suppressing/allowing). If one believes in the idea of *brain rhythms organizing memory retrieval*, then a specific pattern on the 3-torus might correspond to a memory being consolidated or retrieved (some alignment or path shape as evidence).

**Visualization Approaches for $T^3$:** As noted, direct depiction is non-trivial. We have these options to **render** a 3-torus: 1. *Animated 2-torus slices:* Show a regular torus (2D surface in 3D) but animate it or change it over time to represent the third angle. The user's eye then has to integrate the changes to infer the 3D path. 2. *Multiple small multiples:* Show several torus images side by side, each at a successive value of ψ (third angle). Like a flipbook of the 3-torus. This could be static if we choose discrete slices (e.g. delta phase = 0°, 90°, 180°, 270°) and show four tori in a row – differences between them indicate 3D structure. However, that's more analytical and might overwhelm casual users. 3. *Augmented 3D object:* Represent a 3-torus as a shape made of points or as surfaces in our 3D scene. For instance, use particles such that their coordinates (x,y,z) are a function of (φ, θ, ψ) on a 3-torus embedding. Then allow the user to rotate and inspect this point cloud. With interactive 3D (Three.js), this is plausible – the user could literally rotate a cloud or object to see structure. We have to ensure the embedding preserves features: one common embedding for n-torus is using n sine/cosine pairs. For 3-torus, a basic embedding in 4D is: $X = (R + r\cos θ + q \cos ψ)\cos φ$, $Y = (R + r\cos θ + q \cos ψ)\sin φ$, $Z = r \sin θ + q \cos ψ$, $W = q \sin ψ$ (just one idea with an extra radius q for third loop). Then drop W or map W to color. This essentially treats the third circle as an "orthogonal" direction that we can't fully show. We might encode the third angle as color or as an offset of a second torus drawn around the first. E.g., draw a little ring (for third dimension) at the moving point on the main torus whose rotation indicates ψ. In fact, using *orbiting torus* metaphor: have a small torus that orbits around the main torus, where the orbit position = ψ. The small torus itself could even carry some information (maybe its orientation or something encodes something). This way, the user sees a main doughnut and a little doughnut going around it – a bit whimsical, but it's one way to literally add another circle.

However we do it, user testing is crucial to see if people can interpret a 3-torus display. Perhaps they will prefer focusing on two dimensions at a time – we could allow them to freeze one angle to slice or lock onto certain alignments and then examine the remaining degrees of freedom.

**Deliverables for Stage 3:** We will articulate a few strong conceptual mappings for the 3-torus (like the internal/external/task loops model), explain how these could reveal new insights (new expressive power), and provide sketches for visualization (like diagrams of orbiting tori or stacked shells). Given that our codebase already hints at multi-ring structures (the "Levogyre stack" of personal→Earth→Solar shells [57] ), we can align with that: e.g. one shell = personal rhythms (fast), next = environmental (medium), next = cosmic or circadian (slow). A 3-torus could be approximated by three nested tori of different scales representing these, all coupled by some linking (like lines connecting them or one driving the rotation of the next). These sketches and descriptions will serve as a blueprint for implementing a 3-torus mode when the core 2-torus is stable. We'll also note that while the mathematics extend naturally, we must be cautious in UI: any 3-torus mode will likely be an *advanced view* or analysis tool rather than default, due to complexity.

## Stage 4 — 4-Torus, Discrete Time, and Quantum-Inspired Geometry

At Stage 4, we venture into even higher-dimensional tori ($T^4$ and beyond) and introduce ideas from quantum gravity and discrete spacetime. These are highly speculative for a mind interface, so our aim is exploratory blueprints: how might these concepts inform our model of mind-in-time, and what metaphors or structures could they inspire in the UI?

### The 4-Torus ($T^4$)

A 4-torus has four independent circular dimensions. One way to think of it is adding yet another oscillatory factor on top of the 3-torus. In practical cognitive terms, it's challenging to justify four separate inherent cycles unless we are including something like a "meta-rotation" that is not just another brainwave. The prompt suggests "phase × phase × phase × meta-time" as an interpretation. This could mean: three of the circles are like the 3-torus (e.g. three brain rhythms), and the fourth is a *higher-order cycle* – perhaps the cycle of *cycling itself*. Another interpretation: if 3-torus was personal/internal vs external vs task, a 4-torus might add a *social or intersubjective rhythm*, or a *global context rhythm*. For instance, the fourth could be the day/night cycle (circadian ~ 24h, which is definitely periodic and affects cognition). So one could imagine φ, θ, ψ as fast rhythms, and the 4th (χ) as a *daily cycle phase*. Then a 4-torus point encodes what phase of the day it is along with your faster brain rhythms – perhaps enabling patterns like "afternoon slump" vs "morning alertness" to appear as structural differences on the torus. This is quite high-level, but it shows one use: integrate *contextual periodicity*.

Another usage: If we think of quantum or discrete time ideas, a 4-torus could incorporate *discrete phase steps* as an extra dimension. For example, if one treats time not as continuous but as ticking in small increments (like frames or moments), then one could accumulate a phase for each tick relative to a larger cycle. It's abstract, but maybe something like how many theta cycles have passed modulo some number gives another angle (like a cycle count wrap-around). This could relate to *cognitive sequencing*, say every N cycles something resets (like a short-term memory buffer clears every ~7±2 items, just speculating that as a "cycle").

**Projection and Visualization:** A 4D torus cannot be directly visualized, so we rely on projection strategies. We could project 4-torus down to 3D (which we already have to do to show on screen). For example, fix one angle or treat one angle's influence as subtle changes to a 3D shape. Maybe a 4-torus can be seen as a family of 3-tori (just as a 3-torus was a family of 2-tori). Perhaps a visualization could let the user "scroll" or rotate through that fourth dimension. Another possibility is to incorporate the fourth dimension by *colour or animation* while showing a 3D embedding of a 3-torus. E.g., as the 4th angle changes, the colour of a trajectory changes from red to blue cyclically. That means color becomes a circular indicator, effectively adding a visual circle (hue is cyclical). This would embed $T^4$ in $R^3$ with color (or other glyph encoding) as the fourth channel.

**Risks vs. Benefits:** Clearly, a 4-torus is likely overkill for most practical analysis – it's hard enough for users to understand 3 cycles, let alone 4. The risk is information overload and a model so complex it ceases to be useful. However, the benefit might lie in *conceptual exploration* – thinking in 4-torus terms could guide how we structure multi-scale time features. For example, perhaps 4-torus is a stepping stone to thinking about discretized time blocks (next section on CDT). Also, from a purely mathematical standpoint, a 4-torus has richer symmetry and more complex resonance structures; if attention truly had a deep structure (some theorists speak of "the rhythm of timeless emergence" in attention [58]), maybe a 4-torus is needed to capture it. But we will label anything involving $T^4$ as **speculative/metaphorical** unless strong evidence emerges.

## Causal Dynamical Triangulation (CDT)

CDT comes from quantum gravity research and models spacetime as a series of discrete slices ("triangulated" pieces) that maintain causal order [59]. Translating this to our context: we might think of *subjective time* or *cognitive sequences* as composed of discrete "moments" (time quanta) that are connected in a causal chain (one thought leads to another). If our torus models so far have been largely *continuous* (smooth phase flows), CDT is a reminder that perhaps at some fine scale, time might be better seen as a series of *steps* or frames (especially in digital computing contexts or discrete brain states like neuron spikes).

How could we use this idea? One approach: integrate a layer in our visualization where time is depicted as *stacked slices* (like a deck of cards, each slice being the state of the torus at a given moment or short interval). This is essentially creating a cylindrical or tower view: each slice could be an instantaneous phase configuration (for example, a circle representing all current phases, like the state of all oscillators at t; then next slice is state at t+Δ). If we connect these slices with lines (like how in CDT simplices connect points across slices), we create a 3D structure reminiscent of spacetime. The torus structure might emerge as we connect periodic points across slices. For example, if an oscillator has period T, then state at time t and t+T will align, creating a closed loop through the stacked slices. This is a discrete way to identify periodicity: find loops in the stack. In fact, a persistent homology approach on this 3D stack might identify a torus by finding a tunnel, akin to how CDT builds topology from simplices.

We could incorporate CDT-inspired visuals by showing *time as layers of a spiral or torus*. Perhaps a *helix* drawn through successive torus states – if that helix closes perfectly after N slices, that's a periodic orbit (like a rational winding on a torus). The idea of *causal ordering* could reflect that certain events (like cognitive events – decision, insight) only occur when certain phase relationships align. CDT's notion of summing over geometries might parallel us *exploring multiple possible phase configurations*, keeping options open.

In simpler terms for UI: We might present a "time slicing" mode where the user can see discrete frames of their brain state and how one transitions to the next – a bit like a flipbook or comic panels of the torus, where connections show evolution. This could help in understanding state transitions (like how a mind goes from focused to distracted – maybe visible as a certain phase slip propagating).

## Loop Quantum Gravity (LQG) Metaphors

Loop Quantum Gravity posits that spacetime is made of discrete loops (spin networks) at the smallest scale [60] [61]. As a metaphor for cognition, we might imagine that *thoughts or attention states are built from* *discrete relational units* – possibly represented as networks or loops. The theoretical notes speculated: "attention as a coarse-graining over discrete relational graphs (spin networks)" [60] [62]. In our context, one could imagine an **attention graph** where nodes represent moments or chunks of information and links/loops represent relationships or repeated cycles among them.

How to bring that into our system? Perhaps by overlaying a network on the torus: e.g. certain phase configurations (points on torus) are marked as "nodes" if they recur, and transitions between them form links. Over time, you'd get a graph of recurring states (like a state transition graph). This is analogous to extracting a *network of loops* from continuous dynamics. If done at an abstract level, it might reveal patterns like "whenever I enter this focus state (node A on torus), I tend to loop a bit and then either go to node B (distraction) or stay in A for a longer loop."

Another element: **loops as minimal cognitive cycles.** LQG emphasizes loops carrying quantum info. For us, a *loop* could be one complete cycle of attention (say you oscillate from broad to narrow focus and back). Identifying these loops and treating them as fundamental building blocks might be useful. For example, maybe a person has a ~90-second ultradian micro-cycle of focus (just hypothetical). If we detect a loop of that length, we could mark it as a quanta of attention. Then maybe string together multiple loops to form larger tasks, etc. This becomes a kind of *quantum of work* concept. It's a stretch, but practically we could implement a feature to detect repeating patterns in the phase data and highlight them (like "you just completed a focus cycle!").

**Granular vs. Continuous Time:** Combining these quantum ideas with continuous torus: We likely will end up with a hybrid view. Perhaps the underlying processes are continuous oscillations (as EEG suggests), but higher-order constructs like "moments of thought" are discrete. We can visualize granular time by e.g. having discrete markers on the torus each time a certain phase pattern occurs (like beat markers on a record). If attention has a natural beat, we put ticks on the torus circumference to mark it. Over time those ticks might shift if the beat changes. This introduces a quantization: you start counting cycles (one, two, three… reset). If those counts matter (like every 10 cycles we need a break), it becomes reminiscent of time crystals or tick-tock clocks in the mind.

**Bridging Continuous EEG with Discrete Models:** We might consider *coarse-graining*: aggregate many fast cycles into one chunk. For instance, group gamma oscillations by where they fall in slower cycles – that grouping could be considered a discrete event (like one theta cycle containing a certain pattern of gamma = one cognitive operation). In UI, one could have an option to "coarse grain" the view – e.g. compress each theta cycle to a single symbolic element in a sequence. That's moving from analog waveform to digital steps, analogous to going from smooth spacetime to a lattice.

**Visual Hybrid:** Imagine our torus running in real-time (continuous), but behind it, we accumulate a *simplified timeline of discrete blocks*. Each block might represent a stable phase-lock period or a completed cycle. This timeline could be drawn as a series of connected triangles or simplices (CDT style) just below the torus. It might look like a strip of colored triangles, each color encoding the dominant rhythm at that period (like alpha-dominant vs theta-dominant states). If the torus is the analog clock, the CDT strip is the digital log of states. The user could toggle to see this for a retrospective summary ("in the last 5 minutes, you had 3 focus cycles and 2 mind-wandering episodes").

In terms of *quantum metaphors*, our theoretical foundation also mentions the **wave-collapse metaphor**: attention choosing a branch of possibilities [63] [64]. We could incorporate that by showing multiple ghost trajectories on the torus (representing multiple potential futures of phase) and then highlighting the one that actually happens – akin to a quantum superposition collapsing. For example, if the system predicts "you might either maintain this rhythm or drop into a different one," it could visually fan out two paths on the torus; as time goes on, one path's opacity increases (the realized one) and the other fades. This is speculative UI, but it connects the idea of many possible trajectories (like quantum paths) collapsing into one realized attentional path.

**Deliverables for Stage 4:** We will produce a set of *geometrical blueprints* that overlay these advanced concepts on the brainwave data. Expect a few sketches or conceptual diagrams, such as: - A diagram illustrating a 4-torus idea: e.g. three circles for EEG bands and a fourth for day-night cycle, maybe drawn as a torus whose parameters slowly change over 24h (like showing two tori: one at morning, one at night, to imply a 4D shape). - A conceptual figure of *time slices of a torus* to illustrate CDT: stacked tori or a torus sliced into triangular facets connecting across time. - A representation of a *spin network of attention*: maybe a network graph drawn on the torus surface, connecting points that frequently repeat. - Notation of what is firmly based in data vs. what is metaphorical. For example, we note that using circadian phase as an extra angle is empirically grounded (lots of data on cognitive performance vs time of day), whereas treating "quantum collapse" of thought is metaphorical – a playful conceptual tool rather than a tested phenomenon [63].

We will clearly label sections like "CDT-inspired discrete time layering (Speculative)" and "LQG and spin networks (Highly speculative metaphor)" to maintain intellectual honesty, as emphasized in our theoretical notes to stay disciplined about known vs. speculative [65] [66]. The goal here is to expand the vision, planting seeds for future research or design, even if implementation of these in real-time is far off.

## Stage 5 — Implementation & Real-Time Constraints

After all the lofty geometric ideas, Stage 5 brings us back to *pragmatic engineering*. We want to ensure whatever models we propose can actually run in real-time in a web-based visualization (Three.js/WebGL) with live EEG input (e.g. via a Neurosity headset or similar). This means considering algorithmic efficiency, data handling, and visual responsiveness.

**Real-Time Algorithmic Considerations:**

- **Phase Extraction & Smoothing:** Extracting phase in real-time can be done via sliding FFT or IIR filters + Hilbert transform. We need to pick a window short enough to be responsive (~1-2 seconds for EEG likely) but long enough for stability. We can downsample phase computation to, say, 100 Hz update rate, which is plenty for smooth visuals. Using a lightweight DSP library or custom shader for filtering could offload CPU. Alternatively, since the Neurosity SDK might provide band powers directly, we could approximate phase by tracking zero-crossings in the time domain for simplicity (less accurate but very fast). Once we have phase, we maintain a running phase accumulator (unwrapped) and mod 2π for visualization. Smoothing can be done by a short moving average on the instantaneous frequency (difference in phase between frames), effectively low-pass filtering phase velocity to avoid jitter.

- **Data Throughput:** EEG at 250 Hz with multiple channels and bands can be a lot. We should limit ourselves to perhaps 1-2 channels or a few bands at once for the torus, to avoid overwhelming both the visuals and the CPU. We can preprocess signals to get only the needed features (like band-passed signals). If using multiple channels for connectivity metrics, computing PLV in real-time is okay for a couple of pairs (just an average of complex phase differences over a window), but not for dozens of pairs without vectorization. So we might default to one channel's intrinsic rhythms first (intra-channel coupling), and add cross-channel if needed (e.g. frontal vs parietal alpha phase difference mapped on torus – another mapping idea!). This keeps computation simple.

- **GPU Utilization:** Three.js can handle thousands of points and many objects, but for smooth animation at 60fps we want to minimize heavy CPU-JS computations per frame. We can delegate a lot to the GPU via shaders. For example, drawing the torus itself (as a Mesh) is fine – we just update its rotation or material. If we want to draw a trail of the phase point, we can use a line or particle system. A clever approach is to use a shader to draw the spiral patterns on the torus dynamically. Possibly, we can encode the wave patterns (like toroidal/poloidal spirals) as a texture or formula and let the GPU render it per frame based on a phase uniform we update. E.g., a fragment shader could color the torus by $\sin(n \phi + \text{phaseOffset})$ for toroidal stripes, and we just update phaseOffset each frame to make them rotate [45] [46]. This would be very smooth and offload animation from CPU to GPU.

- **Visual Clarity under Motion:** One issue is that the torus will likely rotate or the camera might move (some designs have the torus rotating slowly for aesthetic or to show 3D). We have to ensure our overlays (text, particles) move correctly with it. For instance, if we place marker lines on the torus for certain phase angles, they need to stick as the torus spins. We can parent all torus sub-objects (markers, glows) to the torus object in Three.js so they transform together. Additionally, using a slightly thicker torus or distinct coloring on inner/outer surfaces can help depth perception in motion.

- **Latency and Jitter:** EEG has some latency from acquisition, and we'll add a bit from filtering. But our use-case is not high-frequency trading; a 100 ms delay is fine. The more important thing is jitter – if the data rate is inconsistent or frames drop, the visualization might stutter. We should buffer a bit and interpolate. The torus path (point movement) can be updated with interpolation if data frames skip. Basically, treat the incoming phase as an asynchronous stream, and use requestAnimationFrame to update visuals at a steady rate, interpolating phase in between if needed.

- **Modular Integration (UI):** We must integrate with the app's architecture (Membrane UI, etc.) [67] [68]. This means making our torus viz a *panel* that can be opened/closed, and using state management for its parameters (which are numerous: toggles for spirals, speeds, colors as seen in `torus.js` constructor [69] [70]). We already have an object with many configurable properties (inner radius, tube radius, spiral toggles, etc.) [69]. We should expose the most important controls to the user in the UI (or have sensible defaults and hidden advanced settings). Performance-wise, adjusting these should not be expensive (just changing a uniform or property).

**Pseudo-Code Snippet (Phase to Torus Coordinates):** To clarify the core math, here's a simplified pseudocode for mapping one channel's two band phases to torus 3D coordinates each frame:

```
// Given phases phi and theta (in radians), torus radii R (major) and r (minor)
let x = (R + r * Math.cos(theta)) * Math.cos(phi);
let y = (R + r * Math.cos(theta)) * Math.sin(phi);
```

```
let z = r * Math.sin(theta);
// Update 3D object position (if using a single point rep):
pointObject.position.set(x, y, z);
```

If drawing a continuous tube or surface, Three.js actually can create a TorusGeometry by itself. But since our torus might be dynamic (we may alter r, R, etc., or even deform it), we might update geometry attributes directly for fancy effects. However, for a basic stable torus, using `THREE.TorusGeometry(R, r, tubularSegments, radialSegments)` is fine. Then we can use material shaders for patterns.

For a **particle trail**: We could push the latest point into a buffer and pop the oldest, rendering as a line strip or small spheres. For efficiency, maybe use a single geometry with vertices updated each frame.

**Design Patterns Scaling Up:**

- **From Simple to Complex:** Start with the simplest case (single point moving on a static torus for one channel's two bands). Ensure that runs smoothly. Then add layers: e.g. enable the spiral texture on torus surface to indicate waves – check performance (should be fine). Next, allow two points if comparing two channels, etc. Each feature can be toggled off if performance lags (e.g. disable particle tail on low-end devices). Provide a "quality" setting maybe to drop fancy effects.

- **Progressive Enhancement:** If user's device is strong (desktop with GPU), we can allow more – like animated glimmer particles (which in code are achieved by creating many sprites or points around torus) [47] [48]. If on weaker device, perhaps skip that or reduce count/density. This can be auto-detected or user controlled.

- **Memory Use:** EEG data streams can be long. We should avoid storing entire history at full resolution. Better to compute needed aggregates on the fly (like don't keep hours of phase data points – if we want to show some historical trend, downsample it heavily). Use circular buffers for any stored trails.

- **Integration with Other Panels:** The torus is part of the "Solar Hologram" overlay which is toggled independently [45] [46]. We must ensure enabling this doesn't block other UI interactions – i.e., keep the Three.js canvas transparent for overlay. In `torus.js`, we set the renderer background to transparent [71] for this reason. We'll maintain that so the torus floats over other UI.

- **What to Implement Now vs Later:** At initial implementation, focus on **Stage 1** functionality: a default 2-torus mapping with maybe one alternate mode (like switching bands). Ensure UI controls for basic parameters (toggle spiral display, adjust rotation speed, maybe switch mapping presets). Leave Stage 3+ ideas (3-torus, etc.) for later once 2-torus is robust. Possibly implement a simplified version of Stage 2 (a couple of preset mapping toggles in UI). The quantum-inspired stuff (Stage 4) is definitely for later exploratory development or purely offline analysis.

**Performance-Aware Recommendations:**

- Start with low segment count torus geometry (maybe 64 segments around, 16 tube segments). That looks smooth enough. Increase only if needed for visual quality on high DPI screens.
- Avoid per-frame heavy computations in JS. Precompute as much as possible (for example, if we know we need sin/cos of evenly spaced angles for something, compute once).
- Use requestAnimationFrame effectively – perhaps tie it to incoming data rate if possible (if EEG is, say, 100 Hz, not necessary to update torus faster than that except for eye candy motion).
- Test on target devices (e.g. Oculus if VR, or just Chrome on laptop) to find bottlenecks.
- Optimize three.js object count – one torus mesh, one point, one particle system max. Too many objects kill frame rate. We can often merge geometries if needed (like if we have to show multiple rings, combine into one geometry if static relative to each other).
- Utilize the fact that this is not a full physics simulation – it's mostly geometry and simple math – so it should be quite feasible at 60 fps if done right.

In summary, Stage 5 ensures that our ambitious toroidal mind model doesn't remain on the whiteboard but actually runs in a browser with live data. We provided pseudocode for core mapping, highlighted where to use GPU vs CPU, and planned a phased feature rollout (basic torus first, bells and whistles later). This grounds the project: while we have higher-dimensional dreams, the near-term focus is a solid, smooth 2-torus visualization that can be interacted with via our existing UI framework.

## Final Synthesis

Bringing it all together, we propose a **developmental path** for torus-based brainwave modeling that progresses in dimensionality and complexity, each step opening new possibilities while retaining what was learned before:

1. **Begin with a 2-Torus foundation:** Implement the default 2-torus mapping (e.g. two key oscillations' phases) with intuitive visuals. This provides the core "Donut of Attention" view – immediately interpretable as cycles and rhythms. Use this to engage users and collect feedback on what patterns are meaningful. At this stage, focus on real-time stability, basic user controls, and ensuring the visualization correlates with subjective feelings (e.g. users report "I felt focused here" and indeed the torus shows a coherent pattern). The 2-torus is our workhorse for early experimentation.

2. **Explore the clouds of 2D mappings:** Once the basic torus runs, we introduce the variety of Stage 2 ideas as toggles or modes. Users (or researchers) can try different band combinations, reference frames, etc., without leaving the 2-torus paradigm. This exploration is crucial – it's like scanning the parameter space for resonant configurations. We expect to prune or identify "sweet spots." For example, perhaps we discover that for most people, an alpha-beta torus gives little insight, but a theta-gamma torus lights up during memory tasks. Such findings will guide us on which mappings to standardize or highlight. The cloud approach also makes the system engaging: it's not one static visualization but a little lab for playing with brain rhythm representations.

3. **Introduce 3-Torus for multi-band insight:** When data and user readiness indicate, we move up to the 3-torus (Stage 3). This likely coincides with users wanting to see more at once ("I've seen how my alpha and theta relate, what about adding delta?"). We must decide when to increase dimensionality based on criteria like: *the phenomena of interest clearly involve three distinct frequencies or processes*

*interacting*. For instance, evidence of a triadic coupling (maybe delta affecting theta-gamma coupling) would justify the 3-torus; or a user use-case like neurofeedback targeting synchronization across three frequency bands. Another criterion: user cognitive load. We don't introduce 3D unless users have mastered interpreting 2D patterns. In practice, this might mean only advanced users or researchers toggle 3-torus mode. The introduction should come with explanatory guides (maybe an interactive tutorial) given the jump in complexity. The reward is new expressive power: the 3-torus might reveal state-layering effects (e.g. "Oh, when I get tired (delta up), my focus-related alpha-theta relationship breaks down, and I can *see* that as the torus pattern changing along the delta axis").

4. **Selective 4-Torus overlays and discrete hybrids (only if needed):** A full 4-torus visualization might never be broadly needed, but certain aspects (Stage 4) can trickle in as overlays or analytical tools. For example, integrating circadian phase or other long-cycle context could be done without calling it "4-torus" explicitly – just color-code data by time of day, effectively adding that dimension in a simple way. Or, incorporate a discrete timeline view (CDT-inspired) alongside the torus to highlight periodicities and state transitions in a quantified way. The quantum metaphor parts (LQG spin networks, etc.) may remain metaphorical unless future research gives concrete methods to detect "attention quanta." So we treat Stage 4 ideas as an **optionally overlay** for storytelling and theory-building, careful to distinguish them from the validated continuous models. If down the line, for instance, someone develops a way to measure discrete "frames" of cognition, we have a conceptual slot ready for it.

Throughout this path, we adhere to a **geometry-first approach**: using shapes and topologies as the language to describe mind dynamics. This has several benefits: - It forces us to maintain *consistency across scales and modalities*: by using circles and tori to represent everything from milliseconds oscillations to hours-long cycles, we ensure a unified view rather than a hodgepodge of unrelated charts. This reflects the theoretical stance that attention is scale-invariant and self-similar across levels [17] [72]. - **How we think in time** becomes visually tangible. Instead of just plotting signal amplitude, we're mapping the structure of thought (as oscillations) into space. This spatialization of time may help users understand and even manipulate their sense of time – e.g. noticing when they are stuck in a loop versus when they break out (the torus trajectory pattern would differ). By interacting with the visualization (say, using a breath to intentionally modulate a phase), they get immediate feedback on a geometric object, which is arguably easier to grasp than raw time-series values. Geometry can serve as an "operational language for thinking," as our design ethos suggests, because it externalizes abstract relationships into something one can see and potentially play with. - The toroidal models encourage us to consider *holistic states* rather than single metrics. This is important for consciousness and complex cognition: it's not one frequency or one region, but the interplay. A geometry-first approach captures interplay inherently (loops linking loops). - By gradually increasing dimensionality, we avoid prematurely collapsing to a simplistic model. At each stage, we keep multiple possibilities open (a cloud of 2D models, a few candidate 3D interpretations, etc.), which mirrors the mind's own way of entertaining multiple hypotheses (our UI even encourages ambiguity tolerance and not forcing collapse [73] [74]). In effect, our modeling approach itself is *toroidal*: looping through expansions and refinements rather than a linear path.

In conclusion, the torus-centric framework offers a unifying scaffold to visualize the rhythms of mind and their couplings. Starting with a minimal 2-torus for intuitive clarity, we then open up richer variations and higher-dimensional extensions, guided by both data and user intuition. We only step up a dimension when it reveals qualitatively new insights (e.g. adding a dimension to capture something previously un-representable). By Stage 4, we have a vision that touches deep theory (discrete vs continuous time, quantum analogies) – not to claim the brain *is* a quantum gravity system, but to borrow those well-developed formalisms as inspiration for structuring our thoughts about thought. The final result is not a single static model but a **flexible geometric "language"** for brainwave dynamics. It helps model *how we think in time* – by making time's structure (cycles, loops, phases) explicit – rather than just measuring how signals fluctuate. Ultimately, this geometry-first lens could foster new kinds of interaction with our own minds, turning subjective time from an invisible flow into a navigable landscape of loops and patterns that we can observe, learn from, and gently influence [15] .

---

[1] The Perspective on Time – Is It Linear or Cyclical?
https://www.theperspective.com/debates/living/perspective-time-linear-cyclical

[2] [3] [29] [30] Ensemble of coupling forms and networks among brain rhythms as function of states and cognition | Communications Biology
https://www.nature.com/articles/s42003-022-03017-4?error=cookies_not_supported&code=d4312881-cc2d-4417-ab18-015bcef3b55a

[4] [5] [6] [7] [35] [49] How Do Brain Rhythms Support Cognition? — BrainPost | Easy-to-read summaries of the latest neuroscience publications
https://www.brainpost.co/weekly-brainpost/2024/6/11/fr0hkf1ku0zz71ful0tz5h8dpo9k80

[8] [9] [27] Assessing Connectivity with Phase Locking Value - Sapien Labs | Shaping the Future of Mind Health
https://sapienlabs.org/lab-talk/phase-locking-value/

[10] [11] EEG Phase Slips Reveal Detailed Brain Activity Patterns of Novice ...
https://www.biorxiv.org/content/10.1101/2025.09.03.674031v1.full-text

[12] [13] [14] [25] [26] [51] [52] BULLSEYE_NOTES.md
file://file_000000003648722f9fe6cd99ad4ab108

[15] [16] [17] [18] [19] [20] [21] [22] [23] [24] [43] [44] [55] [56] [57] [58] [60] [61] [62] [63] [64] [65] [66] [72] [73] [74] THEORETICAL_FOUNDATIONS.md
file://file_00000000e050722f8b4c00d55b7903c3

[28] Kuramoto Model for Excitation-Inhibition-Based Oscillations
https://link.aps.org/doi/10.1103/PhysRevLett.120.244101

[31] [32] [33] [34] The Hopf whole-brain model and its linear approximation | Scientific Reports
https://www.nature.com/articles/s41598-024-53105-0?error=cookies_not_supported&code=9c3384f1-98b8-47cf-937f-1968dfd60146

[36] [37] [38] [50] (PDF) EEG spectral attractors identify a geometric core of resting brain activity
https://www.researchgate.net/publication/374733680_EEG_spectral_attractors_identify_a_geometric_core_of_resting_brain_activity

[39] [40] [41] [42] The Lyapunov Exponent in EEG - Sapien Labs | Shaping the Future of Mind Health
https://sapienlabs.org/lab-talk/the-lyapunov-exponent-in-eeg/

[45] [46] [53] [54] [67] [68] UI_ARCHITECTURE.md
file://file_00000000c8f8722f8250b0e06305a21d

[47] [48] [69] [70] [71] torus.js
file://file_00000000069c722fb5adc432280697ed

[59] Extrinsic curvature in two-dimensional causal dynamical triangulation
https://link.aps.org/doi/10.1103/PhysRevD.94.064014