From Synchrony to Structure

Coordination Geometry as a General Principle

of Adaptive Computation

Flyxion

Independent Researcher

2026

Abstract

Three recent developments in unconventional computing, taken together, suggest a common computational principle. In neuromorphic systems, oscillatory synchrony between network nodes enables concept drift detection through changes in coordination patterns before those changes are visible in measurement statistics. In fluid antenna systems, physical communication structures are continuously reorganized by generative reinforcement learning, with port selection and beamforming weights jointly optimized to match changing propagation geometry. In analog compute-in-memory hardware, block-sparse random routing with input-balanced sign encoding achieves near-dense performance while reducing inter-core traffic by over 97%, with optimal effective fan-in remaining approximately constant as network scale grows. This essay argues that all three systems may be fruitfully understood as instances of the same principle: adaptive intelligence emerges from the continual reorganization of relational geometry rather than the manipulation of fixed representations. We formalize this perspective through the Coordination Geometry Framework, in which a system's adaptive capacity is measured by its ability to reorganize $\mathcal{G}(X)$ — the structure of relationships it maintains over its state space X — in response to environmental change. Four claims motivate the analysis: coordination-first detection, representation as coordination residue, sparse coordination sufficiency, and small-world scaling. The RSVP field triple $(\Phi, \mathbf{v}, S_{\text{RSVP}})$ provides a unified field-theoretic vocabulary with per-input synchrony identified with the repair ratio ρ and adaptation with lamphrodyne relaxation. The essay closes with a conjecture: computation is not fundamentally the manipulation of symbols or parameters but the continual reorganization of coordination geometry under constraint.

Computation is not the manipulation of symbols under rules.
Computation is the continual reorganization of coordination geometry under constraint.

Three Systems, One Candidate Principle

Consider three recently proposed adaptive systems that appear, at first reading, to have nothing to do with each other.

The first is a neuromorphic learning architecture in which connection weights are replaced by oscillatory links whose coupling strengths vary rhythmically [1]. Learning occurs by adjusting the phases of oscillators rather than static weights. When the statistical structure of the input changes, the synchronization pattern changes as well, revealing distributional shifts before conventional anomaly detectors notice them. The system detects concept drift not by comparing distributions but by monitoring how its internal coordination reorganizes around the inputs.

The second is a fluid antenna system in which the physical geometry of a communication array is continuously reorganized by a generative-adversarial and reinforcement-learned controller [2]. Instead of fixed antenna elements, the system dynamically selects which ports to activate and what weights to apply, jointly optimizing both decisions to match the changing propagation environment.

The third is an analog compute-in-memory architecture called ScRRAMBLe [3] in which large neural networks are implemented on crossbar arrays of non-volatile memory by imposing block-sparse random routing between computational cores. Rather than dense connectivity, ScRRAMBLe uses sparse inter-core routing tensors with an input-balancing constraint that encodes signed weights using only positive conductances. Networks with as little as 10–25% connection density match the performance of fully-connected counterparts while reducing inter-core traffic by over 97%. Crucially, the effective fan-in — the average number of incoming connections per core — remains approximately constant as the number of cores grows, a signature of small-world organization.

In the first system, computation proceeds through phase relationships among coupled oscillators [6, 7, 8]. In the second, through dynamic selection among physical configurations. In the third, through block-sparse routing geometry among memory cores. The three systems operate in entirely different domains, yet they share a structural pattern suggestive of a common principle: the relevant computational object is not a state but a geometry of coordination, and adaptation is not parameter adjustment but geometric reorganization. This essay attempts to make that pattern precise, develop its consequences, and assess how far the parallel extends.

The Legacy of Fixed Computation

The dominant paradigm in computation rests on an assumption almost never made explicit: that the computational substrate is fixed while the state it stores varies. A circuit computes through fixed gates. A neural network computes through fixed weights. A communication system transmits through fixed antenna elements. A memory accelerator stores through fixed crossbar connections. The architecture is invariant; the content is variable.

This assumption makes computation tractable, but reveals its limitation in changing environments. When the data distribution shifts, the propagation channel changes, or the problem geometry evolves, a fixed substrate optimized for prior conditions may fail. The standard response is retraining: detect the change, discard the old substrate, learn a new one. This is the cycle of fix, break, refix.

Cybernetics already identified this limitation: a system maintaining stable behavior in a variable environment must have sufficient internal variety to absorb the variety of the environment [27]. If the substrate can reorganize continuously, adaptation becomes an ongoing process rather than a periodic crisis.

Definition 1 (Fixed versus reconfigurable computation)**.** A *fixed computation* is a mapping $f_W : X \to Y$ parameterized by a static structure W (weights, topology, physical configuration). Adaptation requires changing W, treated as an exceptional event.

A *reconfigurable computation* is a mapping $f_{\mathcal{G}(t)} : X \to Y$ parameterized by a dynamically evolving coordination geometry $\mathcal{G}(t)$. Adaptation is continuous: $\mathcal{G}(t)$ reorganizes in response to the input stream, and the mapping f changes accordingly.

The distinction is not merely the presence of parameter updates. Gradient descent updates W at each step but is still fixed computation in the relevant sense because the architecture does not reorganize its topology. Reconfigurable computation means that the structure of relationships among computational elements changes: which elements are coupled, how strongly, and in what configuration.

Rhythmic Sharing: Computation Through Phase

The Rhythmic Sharing architecture replaces static connection weights with oscillatory links [4], inspired by astrocytic oscillatory dynamics [5]. The phase matrix Φ evolves according to:

$$\frac{d\Phi}{dt} = \omega_0 + (\epsilon_1 + \epsilon_2 \hat{Q}^T \mathbf{n}) \circ \sin(\Psi - \Phi + \gamma),$$

where Ψ is the local mean phase field, $\hat{Q}$ is the incidence matrix, ϵ_1, ϵ_2 are coupling hyperparameters, γ is a phase bias, and $\circ$ denotes element-wise multiplication. The system uses an Echo State Network substrate [10] in which recurrent dynamics provide a rich manifold of transient states.

The global Kuramoto order parameter

$$R(t)\, e^{i\langle\Phi\rangle(t)} = \frac{1}{N_l} \sum_{k=1}^{N_l} e^{i\Phi_k(t)}$$

measures phase coordination across all links [8]. Per-input synchrony refines this by measuring coordination locally around each input channel i, acting as a microscope for channel-specific drift.

The empirical finding is that per-input synchrony is a sensitive early indicator of distributional change: synchrony for a drifting channel drops before value-level statistics cross the conventional anomaly threshold. The coordination is failing before the failure is visible in the measurements.

Proposition 1 (Coordination-First Detection)**.** Let $X(t)$ denote an environmental signal and $\mathcal{G}(t)$ the coordination geometry maintained by an adaptive system. If environmental change alters the relational structure of X before it alters its marginal statistics, then perturbations in $\mathcal{G}(t)$ will precede perturbations in detectors operating directly on $X(t)$. Consequently, monitoring coordination geometry provides an earlier indicator of system-environment mismatch than monitoring measurements alone.

This proposition is instantiated, but not proved in full generality, by the Rhythmic Sharing results. It functions here as a motivating hypothesis.

Synchrony as an entropy-reducing coordinate

The per-input synchrony variable may be interpreted as a local repair ratio. Let $s_i(t) \in [0, 1]$ denote synchrony around input channel i. Define:

$$S_i(t) = -\log(s_i(t) + \varepsilon),$$

where $\varepsilon > 0$ prevents singularity at zero synchrony. If a drift event reduces synchrony from s_i to $s_i - \delta$ with $0 < \delta < s_i$, then the entropy change is:

$$\Delta S_i = \log\left(\frac{s_i + \varepsilon}{s_i - \delta + \varepsilon}\right) > 0.$$

A synchrony drop is therefore formally equivalent to an entropy-accessibility increase. In RSVP terms, this is precisely the transition toward coordination failure. The detector does not observe the underlying drift directly; it observes the induced increase in S_i.

Fluid Antennas: Computation Through Reconfiguration

The fluid antenna system abandons the fixed-position constraint of conventional arrays [14, 15]. The active geometry at time t is:

$$\mathcal{P}(t) = (A(t),\, W(t)),$$

where $A(t) \in \{0,1\}^{N_s}$ is the binary port activation vector and $W(t) \in \mathbb{C}^{N_s}$ is the beamforming weight vector. The adaptive objective is:

$$\mathcal{P}^*(t) = \arg\max_{\mathcal{P}}\; P(\mathcal{P},\, H(t)),$$

where $H(t)$ is the propagation channel and P is the performance functional. The joint optimization is a high-dimensional mixed-integer non-convex problem solved via a generative reinforcement learning framework (PD-GPPO) combining a primary-dual network for beamforming with generative proximal policy optimization for port selection [2].

The reorganization dynamics operate at two coupled timescales: beamforming weights $W(t)$ reconfigure rapidly given a fixed port selection, while port activation $A(t)$ reconfigures on the episode timescale. Together they constitute a continuously renegotiated coordination geometry between transmitter and propagation channel.

ScRRAMBLe: Coordination Through Sparse Routing

The ScRRAMBLe architecture [3] reveals a third instance of the coordination geometry principle, this time at the hardware level. Standard analog compute-in-memory accelerators face a scaling problem: dense connectivity between cores scales quadratically in communication cost. ScRRAMBLe resolves this by imposing a block-sparse routing structure.

Each core divides its neurons into *slots* — subpopulations of fixed size. Cores communicate through a routing tensor C_{ijkm} that specifies sparse slot-to-slot connections sampled at a connection density $p \ll 1$. A critical insight is the input-balancing constraint: for every positively-weighted incoming connection, a negatively-weighted copy from the same source is routed to another slot. This ensures that the input to each core is zero-mean:

$$\sum_j x_j^{(c)} = 0,$$

which allows signed weight computation using only positive conductances. Crucially, the sign is not a property of any individual conductance element; it is a property of the routing geometry.

Proposition 2 (Sign from coordination)**.** Under the input-balancing constraint, the computation $y = W^{(c)}x^{(c)}$ with signed weights $W^{(c)}$ is implemented by the positive-conductance matrix $G^{(c)}$ via the linear relation $y = a \cdot G^{(c)}x^{(c)}$, where a is a scalar and the zero-mean condition $\sum_j x_j^{(c)} = 0$ eliminates the bias introduced by the conductance offset. Consequently, function (signed computation) is not a property of individual elements but of coordination pattern: the same positive conductance participates in positive or negative effective weight depending on its position in the routing geometry.

Proposition 2 is proved directly by the input-balancing algebra of [3]. Its broader implication is that even the most elementary computational property — sign — can be a relational property of coordination geometry rather than an intrinsic property of a component.

Small-world coordination

ScRRAMBLe reveals an unexpected regularity. As network size increases, the optimal connection density p^* decreases, yet the effective fan-in $\eta = p^* N_{\text{cores}}$ remains approximately constant. Networks with 20 cores use $p^* \approx 0.5$; networks with 60 cores use $p^* \approx 0.2$; in both cases $\eta \approx 10$–12.

This suggests that adaptive systems maintain a characteristic coordination radius independent of total scale. New cores do not require dense connectivity to all existing cores; coordination remains localized while still permitting global information propagation — the hallmark of small-world networks [18, 19]. In the coordination geometry framework, adaptive intelligence appears to operate near a regime of constrained connectivity rather than maximal connectivity: intelligence emerges not from connecting everything to everything else but from maintaining sufficient pathways for repair and reorganization while avoiding unnecessary coordination burden.

Routing as CLIO projection

The ScRRAMBLe routing tensor C_{ijkm} functions as a physically realizable CLIO projection operator. The full state space of possible core activations is projected onto a lower-dimensional routing manifold:

$$\pi_C : X \to \mathcal{G}_{\text{sparse}},$$

where $\mathcal{G}_{\text{sparse}}$ is the subspace of coordination geometries compatible with the routing constraints. The environment is not represented directly; only those relations that survive the routing tensor are preserved. This makes CLIO no longer merely a philosophical projection operator but a physically realizable routing architecture instantiated in hardware.

The Coordination Geometry Framework

Definition 2 (Coordination geometry framework)**.** A *coordination geometry framework* consists of:

- A state space X representing the environment the system operates in.
- A geometry space $\mathcal{G}$ whose elements are coordination structures — configurations of relationships among the system's internal components.
- A performance functional $P : \mathcal{G} \times X \to \mathbb{R}$ measuring how well coordination geometry $G \in \mathcal{G}$ serves the system's objectives given environmental state $x \in X$.
- A reorganization dynamics $\dot{G} = \mathcal{R}(G, x, \nabla_G P)$ governing how the coordination geometry evolves in response to the environment and the performance gradient.

A system is *adaptively intelligent* to the degree that its reorganization dynamics $\mathcal{R}$ maintain $G(t)$ in a high-performance region of $\mathcal{G}$ as $x(t)$ changes.

Element	**Rhythmic Sharing**	**Fluid Antenna**	**ScRRAMBLe**
State space X	Sensor stream	Channel H	Input activations
Geometry $\mathcal{G}$	Phase $\{\Phi, \kappa_{ij}\}$	Port config (A, W)	Routing tensor C_{ijkm}
Performance P	Synchrony / F1	Capacity / SNR	Task accuracy
Reorganization $\mathcal{R}$	Phase dynamics	PD-Net + GPPO	Training + sparsity
Coordination object	Per-input synchrony	Beampattern	Slot routing

Conjecture 1 (Coordination geometry as generalization of parameter adjustment)**.** Any adaptive system whose adaptation can be described as parameter adjustment $\dot{W} = f(W, X, \nabla_W L)$ can be re-expressed in the coordination geometry framework by identifying $\mathcal{G}$ with the configuration of relationships among computational elements induced by W. Parameter adjustment is then a special case in which the topology of relationships is fixed and only coupling strengths change, while the coordination geometry framework additionally permits topological reorganization. This embedding from parameter space into geometry space has not been formally established here.

A taxonomy of reorganization types

Not all coordination geometry reorganizations are equivalent. The three systems examined suggest a preliminary classification along three axes.

Topological vs. parametric reorganization. Topological reorganization changes *which* elements are coupled: port activation in fluid antennas, routing tensor structure in ScRRAMBLe, effective coupling topology in Rhythmic Sharing via phase locking. Parametric reorganization changes *how strongly* existing couplings act: beamforming weights, oscillator coupling strengths. The core claim of the coordination geometry framework is that topological reorganization is primary and parametric reorganization is secondary.

Timescale. Rhythmic Sharing's phase dynamics operate at the oscillation period (fast, continuous). Fluid antenna beamforming weights reconfigure on a channel coherence timescale (fast, continuous). Port selection reconfigures on the RL episode timescale (slow, discrete). ScRRAMBLe's routing tensor is fixed at the architecture level and reorganizes only via retraining (slowest, discrete). A system's adaptive capacity depends on which timescales are active: topological reorganization on fast timescales is more powerful but harder to analyze.

Local vs. global reorganization. ScRRAMBLe's block-sparse structure ensures that each routing change is local (one slot-to-slot connection); global performance emerges from many such local changes. Rhythmic Sharing's phase dynamics are local (each link adjusts based on its neighbors) but can produce global synchronization transitions. Fluid antenna port selection is effectively global (the full activation vector changes). The small-world result suggests that local reorganization with sparse global connectivity may be the optimal regime for scalable adaptive systems.

This taxonomy does not change the core argument: all three systems are better described by coordination geometry reorganization than by parameter adjustment alone. But it clarifies that the framework is a family of mechanisms, not a single mechanism, and that different regimes may require different analysis tools.

A minimal variational form of coordination repair

The reorganization dynamics can be written as gradient flow on a mismatch functional. Let:

$$\mathcal{E}(G, x) = -D(G, x) + \lambda C(G),$$

where $D(G, x)$ measures coordination quality between geometry G and environmental state x, $C(G)$ is a structural cost penalizing excessive reconfiguration, and $\lambda > 0$ controls the trade-off between adaptation and stability. A minimal repair dynamics is:

$$\dot{G} = -\nabla_G \mathcal{E}(G, x) = \nabla_G D(G, x) - \lambda \nabla_G C(G).$$

The first term is adaptive pressure; the second is coherence-preserving inertia.

Proposition 3 (Monotonic repair under gradient flow)**.** If x is held fixed and $G(t)$ evolves by $\dot{G} = -\nabla_G \mathcal{E}(G, x)$, then $\mathcal{E}(G(t), x)$ is non-increasing along trajectories.

Proof. By the chain rule,

$$\frac{d}{dt}\mathcal{E}(G(t), x) = \langle \nabla_G \mathcal{E}(G, x), \dot{G} \rangle = \langle \nabla_G \mathcal{E}(G, x), -\nabla_G \mathcal{E}(G, x) \rangle = -|\nabla_G \mathcal{E}(G, x)|^2 \le 0.$$

The mismatch functional decreases monotonically unless the system is already at a critical point. □

Remark 1 (Non-stationary environments)*.* Proposition 3 assumes a fixed environmental state x. In realistic settings, $x(t)$ and $G(t)$ change concurrently. The total time derivative of the mismatch functional then includes an environmental drift term:

$$\frac{d}{dt}\mathcal{E}(G(t), x(t)) = -|\nabla_G \mathcal{E}|^2 + \langle \nabla_x \mathcal{E}(G, x),\, \dot{x} \rangle.$$

The second term can increase $\mathcal{E}$ even as the system descends the geometry gradient, because the target moves while repair is underway. The framework does not guarantee convergence in non-stationary environments; it provides a tracking bound: if $|\nabla_x \mathcal{E} \cdot \dot{x}| \le |\nabla_G \mathcal{E}|^2$, then the net mismatch remains non-increasing. This is the formal expression of the intuition that adaptation must be faster than environmental change. The coordination-first detection result (Proposition 1) is valuable precisely because it extends the window available for repair: by detecting $\Delta S_i > 0$ before performance degrades, the system gains lead time in which the inequality above is more easily satisfied.

Coordination Geometry and Scale

The coordination geometry framework does not assume that coordination occurs at a single scale. A coordination geometry may exist among neurons, among oscillatory subnetworks, among communication ports, among software modules, or among hardware cores. The framework is scale-relative.

Let $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_n$ denote coordination geometries defined at different organizational levels. A higher-level geometry may emerge from the lower-level geometry through a projection:

$$\Pi_{i \to i+1} : \mathcal{G}_i \to \mathcal{G}_{i+1},$$

forming a tower $\mathcal{G}_1 \to \mathcal{G}_2 \to \cdots \to \mathcal{G}_n$.

Coordination detected at one scale need not be visible at another. A local coordination failure may remain invisible globally; a global coherence collapse may emerge from the accumulation of many local failures. The ScRRAMBLe small-world result instantiates this structure concretely: slot-level coordination (within-core dense connectivity) is distinct from core-level coordination (sparse inter-core routing), and the two levels have different optimal densities. The system achieves global performance through local coordination rather than global dense connectivity.

Coordination Before Representation

The standard picture of adaptive intelligence places representation at the center [24]:

$$\text{Perception} \rightarrow \text{Representation} \rightarrow \text{Action.}$$

The coordination geometry framework suggests an alternative, connecting to enactivist and dynamical-systems traditions [25, 23, 26]:

$$\text{Coordination} \rightarrow \text{Representation} \rightarrow \text{Action.}$$

Stable coordination produces persistent patterns that can be read out as representations. Representations are not the foundation on which coordination is computed; they are the residue of coordination that has been sustained.

In Rhythmic Sharing, the "representation" of the input distribution is not a weight matrix but the current coordination geometry: the phase relationships and coupling strengths the network has settled into through exposure to the data. In the fluid antenna system, the actively maintained port configuration and beamforming weights are the system's best current coordination with the propagation environment. In ScRRAMBLe, the authors explicitly treat the outputs of individual cores as population-coded feature vectors maintained by routing structure rather than by dedicated storage. Features persist through coordination among subpopulations, not through dedicated symbolic storage locations.

Definition 3 (Representation as coordination residue)**.** A *representation* in the coordination geometry framework is a stable pattern that persists in a system's coordination geometry $\mathcal{G}(t)$ across perturbations of the input. Formally, a representation is an equivalence class of coordination geometries that produce the same system behavior under the current class of inputs. Representations are not stored; they are maintained.

Proposition 4 (Representations as coordination invariants)**.** A coordination geometry G^* is a representation in the sense of Definition 3 if it belongs to a stable invariant set $\mathcal{I}$ of the reorganization dynamics under the current environmental class:

$$G^* \in \mathcal{I}, \quad \mathcal{R}(G, x, \nabla_G P)\big|_{x \in \mathcal{X}_0} \in T_G\mathcal{I} \quad \text{for all } G \in \mathcal{I},$$

where $T_G\mathcal{I}$ is the tangent space of $\mathcal{I}$ at G, and the invariant set is Lyapunov stable with basin of attraction $\mathcal{N}(\mathcal{I})$.

Remark 2 (Fixed points vs. limit cycles). The generalization from fixed points to invariant sets is necessary because in practice, coordination attractors are often limit cycles or quasi-periodic orbits rather than point equilibria. Rhythmic Sharing's phase-locked states are limit cycles: the phases continue to evolve, but their relative configuration — the pattern that constitutes the representation — remains stable. Proposition 4 accommodates this by requiring only that $\mathcal{R}$ maps the invariant set to itself, not that any fixed point exists within it. A representation is identified with the behavioral equivalence class of the invariant set, not with a particular point on it.

The proposition is a definition-as-theorem: it characterizes what we mean by "stable representation" in the dynamical systems language of the framework, rather than proving from first principles that representations must be stable invariants. The empirical content comes from the specific systems: for Rhythmic Sharing, the claim is that the per-input synchrony values characterizing a stable input regime form a Lyapunov-stable set under the Kuramoto-like dynamics; for the fluid antenna system, that the optimal port configuration for a given channel class forms a stable fixed point of the RL policy; for ScRRAMBLe, that the routing tensor that achieves near-optimal performance for a given task is stable under small perturbations of the input data.

Memory is a coordination geometry that resists perturbation across long timescales; forgetting is the decay of that geometry when maintenance is insufficient. A long-term memory is a repair equilibrium sustained by the neural coordination structures that support it. The same analysis applies to institutional memory, scientific knowledge, and semantic coherence — each is a coordination geometry that must be actively maintained to persist [27, 28]. The free-energy principle offers a related account in which biological systems minimize prediction error rather than store explicit representations [21].

Drift is the event in which G^* ceases to be a stable attractor: its basin of attraction collapses because $\mathcal{X}_0$ has changed. Detection of drift is detection of this collapse — a synchrony drop, a beampattern mismatch — before the nominal performance metric has yet reflected it. Adaptation is the emergence of a new attractor G^{**} suited to the new environmental class.

Multiple realizations of representation

The identification of representations with coordination residues does not imply a one-to-one correspondence between environmental states and representations. Let $\pi : X \to \mathcal{M}$ be the projection induced by a coordination geometry. In general, $\pi(x_1) = \pi(x_2)$ may hold for distinct $x_1 \neq x_2$: the system compresses many environmental configurations into a single representational state. Representations are equivalence classes of environmental situations that induce sufficiently similar coordination patterns.

This many-to-one structure is not a limitation but a requirement for adaptive behavior. A system producing a distinct representation for every environmental variation would fail to generalize. Stable behavior requires compression. ScRRAMBLe makes this explicit: weight sharing across feature dimensions (the slot mechanism) deliberately maps multiple incoming signals to the same stored weights, achieving generalization through structured invariance rather than explicit case enumeration.

RSVP Interpretation

The RSVP field triple $(\Phi, \mathbf{v}, S_{\text{RSVP}})$ provides a continuous field-theoretic language for all three systems under the same description [22]. The identification is not a proof that RSVP uniquely determines the behavior of these systems; it demonstrates that the field-theoretic concepts provide a natural vocabulary for the coordination geometry framework.

Remark 3 (Sign convention). Throughout this section, $S_{\text{RSVP}} = -\log \rho$ where ρ is the repair ratio — the ratio of semantic support to admissibility threshold at a field node. This is not thermodynamic entropy but an admissibility measure: $S_{\text{RSVP}} \leq 0$ characterizes states inside the admissibility manifold ($\rho \geq 1$) and $S_{\text{RSVP}} > 0$ characterizes states outside it ($\rho < 1$). The zero level set is the admissibility boundary.

Definition 4 (Lamphrodyne relaxation)**.** *Lamphrodyne relaxation* is the process by which a system reorganizes its coordination geometry to reduce S_{RSVP} and restore admissibility following a coordination failure. If $S_{\text{RSVP}}(x) > 0$ at some field node, lamphrodyne relaxation drives $S_{\text{RSVP}}(x) \to 0$ by reorganizing $\mathcal{G}$ rather than by changing x directly.

Definition 5 (RSVP-coordination correspondence)**.** Under the identification ($S_{\text{RSVP}} = -\log \rho$, $\Phi = S_{\text{support}}$, $|\mathbf{v}_{ij}| = \alpha(i, j)$), the three systems map onto RSVP field dynamics as follows.

Rhythmic Sharing: Per-input synchrony $s_i(t)$ is identified with $\exp(-S_{\text{RSVP}}(x_i))$. High synchrony is low S_{RSVP}, meaning tight coordination inside the admissibility region. Phase reorganization when synchrony drops is lamphrodyne relaxation.

Fluid Antenna: The active port configuration concentrates Φ: activating a port at r increases $\Phi(r)$, concentrating constraint propagation at that location. Beamforming weights determine the direction and strength of $\mathbf{v}$. Port selection is the search for the configuration minimizing S_{RSVP} across relevant field nodes.

ScRRAMBLe: The routing tensor C_{ijkm} determines which field nodes are connected by

v and with what strength. Sparse routing is a sparse **v** field; block structure is spatially coherent constraint propagation. The input-balancing constraint ensures that the net entropy flux entering each core is zero-mean, maintaining $S_{\text{RSVP}} \leq 0$ locally by construction.

Operational proxies for S_{RSVP}

The RSVP framework is actionable only if S_{RSVP} can be computed online before performance degrades. The following domain-specific proxies are proposed as computable early indicators, distinct from the lagging performance metrics (F1, SNR, accuracy) that confirm failure after it occurs.

Rhythmic Sharing: $S_i(t) = -\log(s_i(t) + \varepsilon)$ where $s_i(t)$ is per-input synchrony. This is already computed by the algorithm and has been empirically validated as a leading indicator in the paper's datasets.

Fluid antenna: A natural proxy is the condition number $\kappa(\hat{H}(t))$ of the effective channel matrix after port projection, $\hat{H}(t) = H(t) \cdot A(t)$. High condition number indicates that the current port configuration poorly spans the channel, i.e., the projection $\pi_{\mathcal{G}}$ is near-singular — a coordination failure that will produce performance degradation before SNR itself drops below threshold.

ScRRAMBLe: A natural proxy is the variance of input sums across cores, $\sigma^2(\sum_j x_j^{(c)})_c$. Under perfect input balancing this variance is zero. Growing variance signals that the routing geometry is failing to maintain its balancing constraint — the coordination is degrading even if task accuracy has not yet declined.

These proxies are consistent with the Coordination-First Detection principle (Proposition 1): each measures a property of the coordination geometry rather than a property of the output, and each is computable in advance of performance collapse.

Proposition 5 (Adaptation as lamphrodyne relaxation)**.** In all three systems, adaptation has the structure of lamphrodyne relaxation:

$$S_{\text{RSVP}}(x) > 0 \implies \text{reorganize } \mathcal{G} \implies S_{\text{RSVP}}(x) \leq 0.$$

In Rhythmic Sharing, the trigger is a synchrony drop; in the fluid antenna system, a performance decline; in ScRRAMBLe, gradient-based reorganization of the routing tensor during training. All three are instances of a system detecting that it has exited its admissibility region and reorganizing to return.

Connecting back to Proposition 4: the collapse of a coordination attractor is precisely the event $S_{\text{RSVP}} > 0$. Lamphrodyne relaxation re-establishes a stable attractor G^{**} with $S_{\text{RSVP}} \leq 0$, completing the adaptation cycle. Systems monitoring S_{RSVP} directly will therefore detect attractor collapse earlier than systems waiting for performance to degrade — which is the Rhythmic Sharing result.

CLIO and Projection Geometry

Both Rhythmic Sharing and the fluid antenna system perform compression of environmental information into a lower-dimensional coordination manifold:

$$\pi : X \to \mathcal{M},$$

where $\mathcal{M}$ is the manifold of admissible coordination states. In Rhythmic Sharing, the projection maps sensor stream $X(t) \in \mathbb{R}^n$ to the synchrony manifold $S(t) \in [0,1]^n$. In the fluid antenna system, the projection maps channel $H \in \mathbb{C}^{N_r \times N_t}$ to the active configuration $\mathcal{P}(t) = (A(t), W(t))$.

ScRRAMBLe generalizes this: the routing tensor C_{ijkm} implements $\pi_C : X \to \mathcal{G}_{\text{sparse}}$ as a physically realizable operator. The full high-dimensional input space is projected onto the sparse subspace of coordination geometries compatible with the routing constraints. This is not a fixed projection — weight sharing across slots means that the projection adapts as the routing geometry is learned — but its topological structure (which cores connect to which) is fixed by the random routing tensor and the sparsity parameter p.

Proposition 6 (Coordination as adaptive CLIO projection)**.** In all three systems, the coordination geometry $\mathcal{G}(t)$ functions as a CLIO projection of the environmental state $X(t)$: a lower-dimensional manifold that amplifies structure relevant to the system's objectives while compressing structure that is irrelevant. The projection is adaptive: as $X(t)$ changes, $\mathcal{G}(t)$ reorganizes to maintain the quality of the projection. This is the key distinction from fixed dimensionality reduction: the mapping is continuously renegotiated with the environment, not applied statically to it.

The adaptive CLIO projection is what allows all three systems to detect or respond to changes that fixed projections would miss. In ScRRAMBLe, the finding that performance is approximately invariant across a wide range of connection densities (from 10% to 80% in the CIFAR experiments) implies that many distinct routing geometries project into equivalent coordination manifolds for the task at hand — a form of projection robustness consistent with Proposition 6.

Coordination, Performance, and Failure

The coordination geometry framework does not claim that every coordination geometry is beneficial. Coordination can become maladaptive.

Let G^* be a stable coordination attractor. Stability alone does not imply optimality. A system may remain trapped in a geometry that was previously successful but is no longer appropriate: this is the coordination-geometry analogue of the overfitting problem. A network may develop a stable phase pattern suited to training data that fails to generalize; a fluid antenna may lock onto a port configuration suited to a prior channel state; a reservoir may settle into a dynamics that was once adaptive but no longer matches the current environment.

The performance functional $P(G, x)$ mediates the trade-off between stability and reconfigurability. Reorganization dynamics must preserve enough continuity to maintain useful coordination while remaining flexible enough to discover new geometries when environmental conditions change. Proposition 3 shows that gradient flow on the mismatch functional monotonically reduces $\mathcal{E}$, but does not guarantee finding a global optimum. Local attractors in coordination space can trap the system.

The central problem of adaptive computation is therefore not maximizing stability or maximizing change. It is maintaining the appropriate balance between persistence and reorganization — between $\lambda \nabla_G C(G)$ (coherence-preserving inertia) and $\nabla_G D(G, x)$ (adaptive pressure) in the variational form.

Toward Reconfigurable Intelligence

The coordination geometry framework, motivated by three concrete systems from different domains, suggests a broader conjecture about adaptive computation. Current intelligent systems operate predominantly on fixed substrates: large language models have fixed weights, base stations have fixed antenna elements, memory accelerators have fixed crossbar connections, and when the environment changes in ways the fixed substrate cannot accommodate, the intelligence fails.

Reconfigurable intelligence operates differently. The substrate reorganizes, and intelligence is the process of maintaining coordination across that reorganization. Each of the three systems examined here instantiates a specific element of Definition 2.

Rhythmic Sharing in silicon: Neuromorphic hardware based on oscillator networks implements Rhythmic Sharing in silicon [11, 12]. The coordination geometry is the physical oscillator coupling state ($\mathcal{G}$), and $\mathcal{R}$ operates on phase relationships rather than stored weights.

Reconfigurable intelligent surfaces: These implement the fluid antenna principle at large scale [16, 17]. The surface geometry is $\mathcal{G}$, the propagation channel is X, and the configuration policy is $\mathcal{R}$.

Scalable analog memory: ScRRAMBLe demonstrates that sparse coordination geometry can match dense representation at a fraction of communication cost — that the routing manifold $\mathcal{G}_{\text{sparse}}$ is functionally equivalent to the full representation space for the relevant tasks while requiring 97% less traffic. This is empirical evidence that $\mathcal{G}$ can be dramatically sparser than X without loss of adaptive capacity.

Reservoir computing: A high-dimensional dynamical system projects inputs onto a rich manifold of transient states, with learning only at the readout layer [11, 9, 13]. Even when $\mathcal{G}$ is fixed, coordination geometry provides adaptive capacity that static mapping cannot; as $\mathcal{G}$ itself becomes adaptive, the capacity grows further.

Active inference: Organisms maintain internal generative models and minimize free energy by acting on the environment [21]. The generative model is a coordination geometry and free energy minimization is lamphrodyne relaxation renegotiating both internal geometry and environmental coupling simultaneously.

Conjecture 2 (Reconfigurable Intelligence)**.** An adaptively intelligent system $\mathcal{N}$ operating in a changing environment $X(t)$ maintains performance by continuously reorganizing its coordination geometry $\mathcal{G}(t)$ to satisfy:

$$S_{\text{RSVP}}(\pi_{\mathcal{G}}(x)) \leq 0 \quad \text{for all } x \in X(t),$$

where $\pi_{\mathcal{G}}$ is the CLIO projection induced by the current coordination geometry and $S_{\text{RSVP}} \leq 0$ is the admissibility condition. When this condition is violated, the system reorganizes $\mathcal{G}$ via lamphrodyne relaxation. Intelligence is the capacity to sustain this reorganization-repair cycle across the range of environmental changes the system encounters.

Implementation Pathways

The coordination geometry framework suggests a generic four-layer architecture for reconfigurable adaptive systems. The common requirement is that the system must expose its coordination geometry as an object of measurement and control, rather than hiding it inside fixed parameters.

The *sensing layer* receives an environmental stream $X(t)$ — sensor data, channel measurements, user behavior, machine telemetry, biological recordings. Its role is not to solve the task directly but to preserve enough temporal and relational structure for downstream coordination dynamics to detect changes that may not yet appear in local statistics.

The *coordination layer* maintains a dynamic geometry $\mathcal{G}(t)$ over the incoming stream: a graph of coupled oscillators, a reservoir state, an attention topology, a port-selection manifold, a routing tensor, a reconfigurable sensor layout. The essential requirement is that $\mathcal{G}(t)$ changes when the relational structure of $X(t)$ changes.

The *reorganization layer* updates $\mathcal{G}(t)$ in response to mismatch — by reinforcement learning, gradient flow, evolutionary search, message passing, Bayesian filtering, active inference, adversarial generation, or local Hebbian updates. The specific method is less important than the fact that the adaptive variable is the system's relational structure, not merely a parameter vector.

The *readout layer* converts the current coordination geometry into task-relevant action: a synchrony score, a beamforming pattern, a motor policy, a routing decision, a stable representation.

A generic implementation:

$$X(t) \longrightarrow \mathcal{G}(t) \longrightarrow y(t), \quad \text{with} \quad \dot{\mathcal{G}}(t) = \mathcal{R}(\mathcal{G}(t), X(t), P(\mathcal{G}(t), X(t))) \,.$$

The practical design rule: do not only ask what state the system is in. Ask what coordination geometry the system is maintaining, how that geometry fails, and what reorganization process restores it.

Open Questions

The coordination geometry framework currently sits at the level of a research programme rather than a fully formalized theory. Three questions mark the boundary between what the framework establishes and what it leaves open.

Falsifiability. The claim that adaptive intelligence is fundamentally coordination geometry reorganization would be falsified by a system that: (a) achieves robust performance across a changing environment without any topological reorganization, using only parametric adjustment of fixed weights; and (b) cannot be re-described as coordination geometry reorganization without making the framework vacuous (i.e., without identifying "geometry" with the weights themselves). Deep learning systems that generalize across distribution shifts via weight interpolation are the strongest candidate counterexample. If interpolation between fixed-weight solutions is sufficient for robust generalization, the topological reorganization requirement may be too strong. The framework would need to be weakened to: "topological reorganization is *sufficient* but not *necessary* for robust adaptation, and it is the most efficient mechanism when it is available."

The small-world constant. The ScRRAMBLe result that effective fan-in $\eta \approx 10$–12 remains constant across scales is striking. Whether this constant can be derived from first principles within the coordination geometry framework is an open problem. A candidate derivation: η is the value at which the repair-rate term $|\nabla_G \mathcal{E}|^2$ in Proposition 3 balances the environmental drift term $|\nabla_x \mathcal{E} \cdot \dot{x}|$ at a typical rate of environmental change. If so, η is not a universal constant but a function of the ratio of reorganization speed to environmental change rate — a predicted scaling law that could be tested by varying the non-stationarity of the input in ScRRAMBLe-style experiments.

A common algorithmic principle. The three systems use different optimization paradigms: phase-coupled ODEs (Rhythmic Sharing), generative RL (fluid antennas), gradient descent on routing tensors (ScRRAMBLe). A candidate unifying principle is online projection pursuit: each system is continuously searching for the low-dimensional coordination manifold that best projects the environmental structure relevant to the task. Rhythmic Sharing finds this manifold through synchronization; fluid antennas through port selection and beamforming; ScRRAMBLe through sparse routing. The three are instances of different algorithms for the same optimization problem: minimize $S_{\text{RSVP}}(\pi_{\mathcal{G}}(x))$ over the space of admissible coordination geometries. Whether this can be given a unified algorithmic treatment — for instance, as geodesic flow on $\mathcal{G}$ equipped with a metric derived from the mismatch functional $\mathcal{E}$ — is the deepest open question raised by the framework.

Conclusion

The Rhythmic Sharing algorithm shows that concept drift is detectable through coordination geometry reorganization before it is detectable in value-level statistics. The fluid antenna system shows that communication performance is improvable through physical geometry reorganization beyond what fixed-substrate optimization achieves. ScRRAMBLe shows that block-sparse coordination geometry can match dense representation at a fraction of the communication cost, and that optimal effective fan-in remains constant as network scale grows — an empirical signature of intelligence emerging from sparse relational structure rather than dense accumulation.

The four central claims of the essay are:

Coordination-first detection (Proposition 1): adaptive systems often detect environmental change through perturbations in coordination geometry before those perturbations are visible in measurement statistics.

Representation as coordination invariant (Definition 3, Proposition 4): representations are stable invariant sets in coordination space, accommodating both fixed points and limit cycles. Memory is coordination maintenance. Forgetting is coordination decay. Drift is invariant set collapse.

Sparse coordination sufficiency (Proposition 2, ScRRAMBLe results): coordination geometries far sparser than the full representation space can preserve adaptive capacity. Intelligence depends on the geometry of admissible coordination pathways, not on maximal connectivity.

Small-world scaling: optimal adaptive systems maintain approximately constant effective fan-in as they scale, operating near a regime of constrained connectivity rather than maximal connectivity.

The RSVP framework provides a unified vocabulary: Φ encodes coordination density, $\mathbf{v}$ encodes its propagation, and $S_{\mathrm{RSVP}} = -\log \rho$ measures admissibility. Drift is $S_{\mathrm{RSVP}} > 0$. Adaptation is lamphrodyne relaxation.

Representation is not the foundation of intelligence but its stable residue. What persists when coordination is sustained is a representation; what dissolves when coordination fails is a representation. The foundation is the coordination itself: the ongoing work of maintaining a geometry of relationships that keeps the system within its admissibility manifold as the world changes.

The map does not precede the territory. The map is what the territory looks like when the coordination between observer and environment has been sustained long enough to leave a stable trace.

Flyxion / Independent Researcher. 2026.

References

References

[1] Whitehouse, I., Kang, H., and Losert, W. "Emergent detection of concept drift within the glia-inspired 'rhythmic sharing' algorithm." *npj Unconventional Computing* 3:25, 2026.

[2] Zhang, J., Xing, C., Tang, J., Yang, Z., Deng, N., Zhao, N., Wong, K.-K., Niyato, D., and Chae, C.-B. "Generative reinforcement learning for fluid antenna systems based sensing and communication." *npj Wireless Technology* 2:30, 2026.

[3] Jaltare, V., Mondal, R., Gibb, L., Song, J., Leugering, J., and Cauwenberghs, G. "ScRRAMBLe: block-sparse deep learning architecture for analog in-memory computing accelerators." *npj Unconventional Computing* 3:26, 2026.

[4] Kang, H., and Losert, W. "Rhythmic sharing: a bio-inspired paradigm for zero-shot adaptive learning in neural networks." *Physical Review Research* 8:013267, 2026.

[5] O'Neill, K. M., et al. "Decoding natural astrocyte rhythms: dynamic actin waves result from environmental sensing by primary rodent astrocytes." *Advanced Biology* 7:e2200269, 2023.

[6] Kuramoto, Y. *Chemical Oscillations, Waves, and Turbulence*. Springer, 1984.

[7] Strogatz, S. H. *Sync: The Emerging Science of Spontaneous Order*. Hyperion, 2003.

[8] Acebrón, J. A., Bonilla, L. L., Pérez Vicente, C. J., Ritort, F., and Spigler, R. "The Kuramoto model: a simple paradigm for synchronization phenomena." *Reviews of Modern Physics* 77(1):137–185, 2005.

[9] Jaeger, H. "The echo state approach to analysing and training recurrent neural networks." Technical Report GMD 148, German National Research Center for Information Technology, 2001.

[10] Jaeger, H., and Haas, H. "Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication." *Science* 304:78–80, 2004.

[11] Maass, W., Natschläger, T., and Markram, H. "Real-time computing without stable states: a new framework for neural computation based on perturbations." *Neural Computation* 14(11):2531–2560, 2002.

[12] Lukoševičius, M., and Jaeger, H. "Reservoir computing approaches to recurrent neural network training." *Computer Science Review* 3(3):127–149, 2009.

[13] Verstraeten, D., Schrauwen, B., D'Haene, M., and Stroobandt, D. "An experimental unification of reservoir computing methods." *Neural Networks* 20(3):391–403, 2007.

[14] Wong, K.-K., Shojaeifard, A., Tong, K.-F., and Zhang, Y. "Fluid antenna systems." *IEEE Transactions on Wireless Communications* 20:1950–1962, 2020.

[15] Wang, C., et al. "AI-empowered fluid antenna systems: opportunities, challenges, and future directions." *IEEE Wireless Communications* 32:166–173, 2024.

[16] Di Renzo, M., et al. "Smart radio environments empowered by reconfigurable intelligent surfaces." *EURASIP Journal on Wireless Communications and Networking*, 2019.

[17] Başar, E., Di Renzo, M., de Rosny, J., Debbah, M., Alouini, M.-S., and Zhang, R. "Wireless communications through reconfigurable intelligent surfaces." *IEEE Access* 7:116753–116773, 2019.

[18] Watts, D. J., and Strogatz, S. H. "Collective dynamics of 'small-world' networks." *Nature* 393:440–442, 1998.

[19] Bassett, D. S., and Bullmore, E. "Small-world brain networks." *Neuroscientist* 12(6):512–523, 2006.

[20] Dalgaty, T., et al. "Mosaic: in-memory computing and routing for small-world spike-based neuromorphic systems." *Nature Communications* 15:142, 2024.

[21] Friston, K. "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience* 11(2):127–138, 2010.

[22] Freeman, W. J. *Neurodynamics: An Exploration in Mesoscopic Brain Dynamics*. Springer, 2000.

[23] Beer, R. D. "Dynamical approaches to cognitive science." *Trends in Cognitive Sciences* 4(3):91–99, 2000.

[24] Clark, A. *Surfing Uncertainty: Prediction, Action, and the Embodied Mind*. Oxford University Press, 2016.

[25] Varela, F. J., Thompson, E., and Rosch, E. *The Embodied Mind*. MIT Press, 1991.

[26] Gibson, J. J. *The Ecological Approach to Visual Perception*. Houghton Mifflin, 1979.

[27] Ashby, W. R. *An Introduction to Cybernetics*. Chapman and Hall, 1956.

[28] Holland, J. H. *Hidden Order: How Adaptation Builds Complexity*. Addison-Wesley, 1995.