Revisiting Boids for Emergent Intelligence via Multi-Agent Collaborative Tool-Building

Xisen Wang*
University of Oxford
xisen.wang@keble.ox.ac.uk

Qi Zhang*
University of Oxford
qi.zhang.agi@gmail.com

Abstract

Can LLM-based agents exhibit emergent intelligence when governed only by simple, local rules—without predefined workflows or central coordination? We explore this question by extending the classical Boids framework from physical flocking to cognitive collaboration, using it to study how multi-agent systems spontaneously organize around tool creation. Unlike prior work that treats tool building merely as a vehicle for downstream task improvement, we frame it as a lens for understanding how decentralized interaction gives rise to coordination, specialization, and long-horizon adaptation. Each agent follows an *observe–reflect–build* loop within a shared environment, guided only by three Boids-inspired primitives—*separation*, *alignment*, and *cohesion*. Through these minimal rules, agents collectively invent, adopt, and refine tools. Our results show that Boids-style coordination sustains long-horizon exploration and diversity in open-ended domains, supporting the continuous accumulation of structural and functional complexity beyond what uncoordinated baselines achieve. Our contributions are twofold: (1) an end-to-end infrastructure and metrics for collective tool building as a sandbox for emergent intelligence; (2) a Boids-inspired algorithm that demonstrates how simple local rules can trigger complex collaborative dynamics and long-horizon complexity growth.

1 Introduction

Emergent intelligence in multi-agent systems has been widely explored through communication protocols, cooperative tasks, and workflow-centric testbeds. Prior work shows how simple agents can coordinate, negotiate, and divide labor under predefined tasks or social dilemmas [1, 2, 3, 4], and how large language models (LLMs) can plan and *use* external tools to accomplish complex goals [5, 6, 7, 8]. However, testbed construction via tool building itself remains largely absent for swarm intelligence research. The few existing efforts treat tool construction only as a supporting mechanism for improving downstream task performance—for instance, by framing LLMs as tool makers [9], disentangling tool design from execution [10], or unifying actions as executable code [11]—rather than as a lens for probing emergent intelligence and multi-agent coordination in their own right.

We argue that tool building is a uniquely revealing substrate: tools are structured, composable artifacts whose collaborative invention and reuse expose modularity, adoption, and ecosystem dynamics in ways that single-task workflows or pure communication cannot. As LLMs advance in coding and reasoning ability, and as agent societies scale, tool creation becomes increasingly central—not only for solving tasks, but also for measuring how decentralized rules give rise to specialization, composability, and long-horizon adaptation.

*Equal contribution. Corresponding Author: xisen.wang@keble.ox.ac.uk

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Scaling Environment of Agents.

To address this gap, we propose an integrated infrastructure that establishes *collaborative tool building* as a first-class arena for systematically studying emergent intelligence and agent communication. Agents follow an *observe–reflect–build* loop over a shared registry, automated test suite, and a *Tool Complexity Index (TCI)* that quantifies code size, interface richness, and compositional sophistication, alongside complementary measures of adoption and reliability. This substrate provides both the medium for collective invention and the instrumentation to measure modularity, specialization, and ecosystem dynamics.

On top of this substrate, we revisit Reynolds' foundational insight that simple local rules—*separation*, *alignment*, and *cohesion*—can generate coherent global structure without centralized control [12, 13, 14]. While originally formulated for spatial coordination in flocks and swarms, these principles generalize to cognitive collectives where agents share ideas rather than positions. We adapt them into a minimal model of *cognitive coordination*: *separation* promotes creative divergence by encouraging agents to explore novel tool directions; *alignment* supports imitation and knowledge transfer across agents; and *cohesion* ensures integrability, sustaining coherence within the shared tool ecosystem. Together, these simple local interactions provide a lens for examining how complex collective behavior—division of labor, reuse cascades, and cumulative complexity—can arise without predefined workflows or explicit planning.

Contributions. This paper makes two contributions: (1) We introduce an end-to-end tool-building infrastructure—from system design to metrics—that establishes a sandbox for collective invention and reveals emergent dynamics. (2) We develop a Boids-inspired algorithm for multi-agent collaboration, showing how simple local rules can trigger complex forms of coordination and long-horizon complexity growth.

2 Tool-Building Infrastructure for Emergent Intelligence

2.1 Agent Substrate and Execution Loop

To study emergent intelligence, we first construct a substrate where tools authored by one agent can be safely adopted and composed by others. The minimal design supports decentralized creation, refinement, and sharing while preserving safety and observability.

Overview and agent loop. Each agent executes an *observe–reflect–build* loop over five components: (i) **Agent Identity** with a light specialization prior (e.g., cleaning, profiling); (ii) a **Shared Tool Registry** recording community-visible artifacts and adoption telemetry; (iii) a **Personal Tool Space** for private drafts and tests; (iv) a **Reflection History** logging observations, choices, and outcomes; and (v) an **Environment Manager** abstracting resources/constraints. At each timestep, the agent scans registry artifacts and test outcomes, reasons about unmet needs and ecosystem gaps, and proposes either a new tool or a targeted refinement. Tools follow a standardized interface (typed signature, structured returns) to enable composition. Execution occurs in a centralized, sandboxed runtime that enforces safety (e.g., recursion/timeout limits) and accrues usage telemetry. This compositional substrate allows dependency chains to traverse agents, providing the medium for emergent collaboration.

Assurance and specialization dynamics. Every proposal triggers automated quality control: *test generation* (cases probing functional coverage), *execution tracking* (pass/fail rates and error logs), *visibility* (propagating outcomes to all agents), and *persistence* (structured logs for longitudinal study). We add light biases to promote division of labor: *Meta-Prompt Influence* nudges broad domains without hard constraints; *Usage-Based Reinforcement* increases visibility and survival of adopted tools; *Failure-Driven Adaptation* focuses agents on systemic test failures by proposing complementary utilities; and *Neighbor Awareness* reduces redundancy by exposing recent peer contributions. Together, assurance and bias produce a feedback loop in which successful tools persist, unsuccessful ones are repaired or pruned, and niches of specialization gradually crystallize.

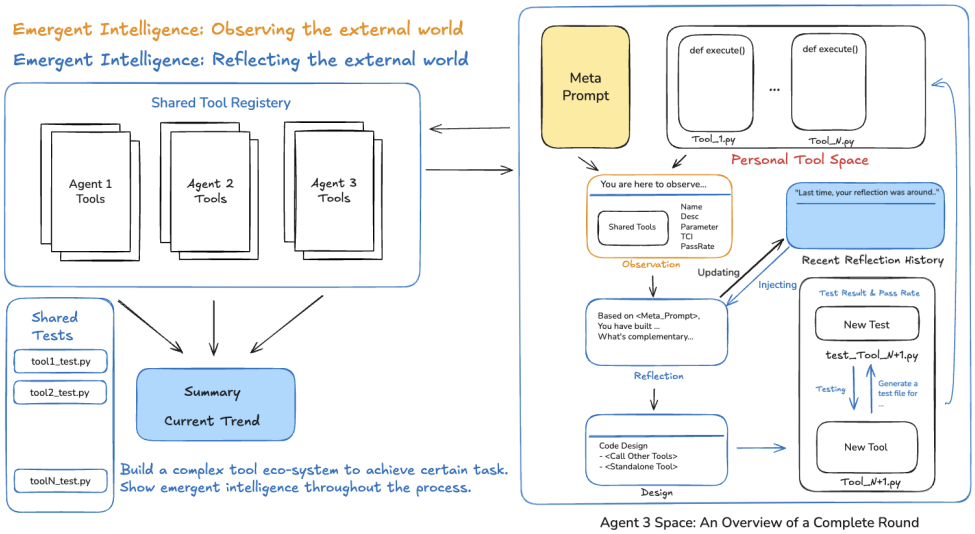

Figure 1: Tool Building Infrastructure Overview: A complete round in the emergent intelligence framework. The diagram shows how an agent observes shared tools, reflects on complementarity and history, designs new tools, and contributes back to the shared registry and test ecosystem.

2.2 Measurement and Study Design

Tool complexity and reliability. We quantify artifact sophistication with a *Tool Complexity Index* (TCI), a five-term linear score that captures structural richness without requiring heavyweight static analysis:

$$\mathrm{TCI}(t) = w_\ell \cdot \mathrm{LOC}(t) + w_\sigma \cdot \mathrm{Interface}(t) + w_\rho \cdot \mathrm{Return}(t) + w_\kappa \cdot \mathrm{Calls}(t) + w_\iota \cdot \mathrm{Imports}(t).$$

Here, $\mathrm{LOC}(t)$ counts non-empty, non-comment source lines; $\mathrm{Interface}(t)$ records the number of typed parameters in the exported `execute` signature; $\mathrm{Return}(t)$ scores the cardinality of structured return fields; $\mathrm{Calls}(t)$ counts explicit intra-registry tool invocations; and $\mathrm{Imports}(t)$ counts non-standard-library imports, after deduplication. The weights $(w_\ell, w_\sigma, w_\rho, w_\kappa, w_\iota)$ are fixed across all conditions and calibrated once on a small validation set to align with human judgments of sophistication. Because raw complexity can be inflated by unused or unrunnable code, we always report TCI jointly with *adoption rate* $\mathrm{Adopt}(t)$—the number of distinct downstream tools that import t—and *reliability* $\mathrm{Pass}(t)$, the fraction of autogenerated tests that execute without error.

Comparative study design. We evaluate these observables under matched agent counts, horizon lengths, and meta-prompts across all experimental conditions. The *baseline* condition employs the unaugmented observe–reflect–build loop; successive variants introduce Boids-style communication rules (alignment, separation, cohesion), self-reflection memory, and other coordination scaffolds one at a time. For each configuration we execute paired runs on both meta-prompts (literary analysis vs. healthcare analytics) and—where relevant—paired model families (GPT-4.1-nano and GPT-4o-mini). Metrics are computed per run, averaged across repetitions, and accompanied by confidence intervals derived from bootstrap resampling of rounds. This matched design ensures that changes in complexity growth, specialization, or duplication can be attributed to the coordination mechanisms rather than shifts in task, population size, or randomness.

Core performance metrics. Each experimental run records performance and behavioral indicators across the multi-agent society, yielding a compact set of comparative metrics summarized in the review table. These metrics capture both absolute performance and relative changes with respect to baseline

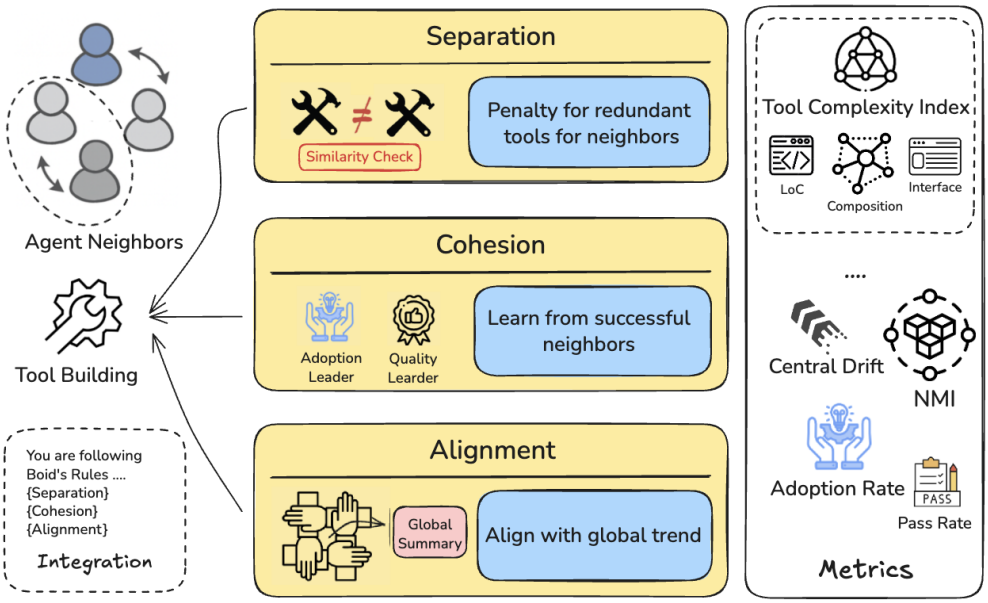

Figure 2: Overview of Boids-based coordination for collaborative tool building.

conditions: (1) Δ*Avg TCI*, the delta in mean Tool Complexity Index relative to baseline, computed as the difference in average TCI scores across all generated tools between the experimental condition and the control; (2) Δ*Pass Rate (pp)*, the absolute change in test success rate measured in percentage points, representing the difference in the proportion of passing unit tests between experimental and baseline conditions; (3) Δ*Category*, the net change in the cardinality of distinct functional categories present in the tool repository, reflecting expansion or contraction of representational task diversity; (4) *Complexity* Δ, the aggregate change in ecosystem-wide complexity, quantified as the difference in cumulative TCI accumulation over the experimental duration; (5) *Agent TCI* $\pm\sigma$, the per-agent mean Tool Complexity Index with standard deviation, characterizing the average sophistication of tools produced by individual agents and the dispersion across the agent population; (6) *Spec* $\pm\sigma$, the degree of agent specialization measured by normalized mutual information $\mathrm{NMI}(A; C)$ between agent identities and task categories, reported with standard deviation to capture variation in specialization patterns; and (7) *Consistency* $\pm\sigma$, operationalized as the inverse coefficient of variation in per-agent TCI distributions, quantifying the uniformity with which agents produce tools of comparable complexity, where lower values indicate greater heterogeneity in individual production profiles.

3 Boids–Inspired Framework for Collaborative Tool

Building Our framework adapts the classical boids model from spatial coordination to the cognitive domain of multi-agent tool creation. The core of an agent's decision-making process is governed by three rules—separation, alignment, and cohesion—which are mathematically formulated and implemented with prompts to guide behavior based on local information within the agent's neighborhood. The three rule outputs are synthesized into textual guidance, which is then incorporated into the agent's reflection process to produce its final decision in tool making.

3.1 Separation: Functional Niche Specialization

The separation rule enforces functional diversity and encourages niche specialization by discouraging the creation of tools that are redundant within an agent's local neighborhood. We model this through calculating redundancy scoring and prompting.

Neighbourhood and textual representations. To encourage niche specialization, each agent inspects the tools proposed in its local neighbourhood and raises a redundancy warning when neighbours converge semantically. Neighbourhoods follow a k-regular ring over a population of size n , so each agent observes its k nearest predecessors and k nearest successors under circular indexing. For every neighbour tool, we build a short document by concatenating its name and description, apply standard tokenization with stop-word removal, and compute Tool Term features. The resulting document is mapped to a nonnegative Tool Term vector and ℓ_2-normalized; we denote the unit embedding of the m-th neighbour tool by $\mathbf{v}_m$.

Redundancy scoring. Semantic overlap between two neighbour tools p and q is measured by cosine similarity,

$$S_{pq} \;=\; \frac{\langle \mathbf{v}_p, \mathbf{v}_q \rangle}{\|\mathbf{v}_p\| \, \|\mathbf{v}_q\|},$$

where $\mathbf{v}_p$ and $\mathbf{v}_q$ are the ℓ_2-normalized Tool Term embeddings of their respective documents, $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product, and $\|\cdot\|2$ the Euclidean norm. Since the embeddings are unit-length, $S_{pq} = \langle \mathbf{v}_p, \mathbf{v}_q \rangle \in [0, 1]$.

Flagging and guidance. With a similarity threshold θ and quota $K = 2$, we rank all neighbour pairs with $S_{pq} \geq \theta$ and collect up to K ***unique*** tools appearing in the top pairs, annotating each with its maximal observed similarity. The agent receives a concise separation message (names, brief descriptions, leading code lines) and an explicit instruction to propose a functionally distinct tool next; no numeric penalties or probabilistic adjustments are applied, and the message is injected into the reflective prompt conditioning the subsequent proposal.

3.2 Alignment: Exemplar-Guided Propagation of Successful Design Principles

The goal of alignment is to propagate effective design principles by exposing each agent to recent, high-quality exemplars created by its neighbours. Consider agent i with a pre-defined k-ring neighbourhood $\mathcal{N}_i$ and a discrete round index r. A fixed recency horizon of $w = 3$ rounds defines the candidate pool $\mathcal{R}_i(w)$: the set of all tools produced by any neighbour $j \in \mathcal{N}_i$ in rounds greater than or equal to $r - w$. Each tool $\tau \in \mathcal{R}_i(w)$ is annotated with three signals: a binary test outcome $\mathrm{test\_passed}(\tau) \in \{0, 1\}$, a non-negative adoption count $\mathrm{adopt}(\tau) \in \mathbb{N}_0$ (number of observed reuses), and a Tool Complexity Index $\mathrm{tci}(\tau) \in [0, 10]$ that scores architectural sophistication. The primary (quality) exemplar is selected by maximizing complexity among recent tools that have passed tests,

$$q^\star \;=\; \arg \max_{\tau \in \mathcal{R}_i(w) \,:\, \mathrm{test\_passed}(\tau) = 1} \; \mathrm{tci}(\tau),$$

where $q^\star$ denotes the chosen exemplar, $\mathcal{R}_i(w)$ is the recency-filtered neighbourhood set defined above, $\mathrm{test\_passed}(\cdot)$ is the binary test indicator, and $\mathrm{tci}(\cdot)$ is the complexity score. If no recent tool has passed tests, the same maximization of $\mathrm{tci}(\cdot)$ is performed over all $\tau \in \mathcal{R}_i(w)$ and the resulting tool is used as a fallback quality exemplar. A secondary (adoption) exemplar is provided only when some recent tool exhibits reuse; in that case, the tool in $\mathcal{R}_i(w)$ with the largest $\mathrm{adopt}(\tau)$ is identified (and omitted otherwise). The mechanism of influence is purely prompt-level: the system assembles a concise segment featuring $q^\star$ and, when available, the adoption exemplar, each with provenance, strategic commentary, and a code excerpt. The agent is instructed to emulate the highlighted principles—e.g., modularity, composition, and robust interfaces—without copying. No numeric action preferences, penalties, or probabilistic selection adjustments are computed; the downstream proposal is produced solely by conditioning the language model on this narrative guidance.

3.3 Cohesion: Trend Alignment via Global Round Summary

The mechanism promotes coordinated progress by steering each agent toward the ecosystem's emerging collective objective inferred from the immediately preceding round. Let $r \in \mathbb{N}$ denote the current round index. At the end of round $r - 1$, we assemble a dataset $\mathcal{A}_{r-1}$ consisting of one

descriptor for each successfully built tool in that round; each descriptor records the authoring agent identifier, the tool's name, and a succinct design synopsis. An external large language model (the "senior architect") operated under a fixed prompting policy $\pi_{\text{architect}}$ then produces a one-paragraph global trend summary and next-step suggestion, denoted $\mathcal{G}_{r-1} \in \mathsf{Text}$, according to

$$\mathcal{G}_{r-1} \;=\; \begin{cases} \mathrm{Summarise}(\mathcal{A}_{r-1};\, \pi_{\text{architect}}), & \text{if } \mathcal{A}_{r-1} \neq \varnothing, \\ \text{'No significant tool-building activity occurred in this round.''}, & \text{otherwise,} \end{cases} \tag{1}$$

where $\mathrm{Summarise}(\cdot;\cdot)$ denotes the application of the senior-architect LLM, configured by the fixed policy $\pi_{\text{architect}}$, to the round-$(r-1)$ descriptors $\mathcal{A}_{r-1}$. The sequence $(\mathcal{G}_0, \ldots, \mathcal{G}_{r-1})$ is retained as the centre-history for use in later rounds. *Mechanism of influence.* At the start of round r, each agent i receives a cohesion prompt comprising a fixed "Cohesion" tag, the textual artifact $\mathcal{G}_{r-1}$, and the instruction: "Contribute to this emerging trend; design a tool that serves the broader goal." This is strictly prompt-level guidance: it augments the agent's reflection context without computing numeric preference weights, penalties, or probabilistic mixing. The subsequent tool proposal is generated by the agent's LLM-conditioned policy using $\mathcal{G}_{r-1}$; when $\mathcal{G}_{r-1}$ equals the fallback sentence above, the agent is explicitly informed that no meaningful trend has yet emerged.

3.4 Decision Synthesis

Prompt-mediated synthesis. For each agent turn, the Boids rules generate natural-language guidance snippets: alignment and separation derive from neighbours' tool metadata, while cohesion injects the previous round's global summary. These snippets are concatenated into the agent's reflection prompt and provided—together with mission context—to the language model.

Implication. Decision-making thus remains conversational: the repeated application of Boids guidance shapes emergent behaviour via the LLM's conditional generation rather than via stochastic policies over engineered utility scores.

4 Experimental Design

Overview. We operationalize a high-throughput, reproducible pipeline for multi-agent *tool creation* that systematically spans coordination regimes, model families, and task contexts while holding ancillary factors fixed. Each run instantiates a cohort of 15 agents for 15 iterative build rounds, yielding $15 \times 15 = 225$ tool implementations per configuration. We evaluate six methodological regimes: (i) the base *reflective* loop, (ii–iv) single-rule Boids augmentations (alignment, separation, cohesion), (v) the full Boids triad, and (vi) Boids+recent-reflection memory. Every regime is executed with two production-grade models (GPT-4.1-nano, GPT-4o-mini), giving

$$225 \text{ tools/run} \;\times\; 6 \text{ regimes} \;\times\; 2 \text{ models} = 2700 \text{ tools per meta-prompt.}$$

We run this factorial grid across two orthogonal meta-prompt contexts, totaling $2{,}700 \times 2 = 5{,}400$ unique tools in the principal study.

Meta-prompt contexts. To stress divergent reasoning modalities under identical infrastructure, we use: (1) a *literary intelligence* brief (Shakespearean sonnet interpretation and stylistic decomposition), and (2) a *healthcare insurance analytics* brief (actuarial reasoning over public medical-cost datasets). Both are curated from public-domain sources to emphasize long-horizon planning and composition rather than dataset idiosyncrasies. Running the same coordination regimes and models in both contexts enables cross-domain validation without confounding changes in evaluation protocol.

5 Results

5.1 Data Science Domain: Coordination as Functional Amplifier

Coordination transforms local interaction into functional amplification. In the data-science domain, *Full Boids* with GPT-4.1-nano achieves the strongest overall performance, combining elevated complexity (+4.8%) with the highest test pass rate (+46%). This dual improvement is non-trivial, as complexity and correctness typically trade off. The *Alignment* condition produces

Model	Method	Avg TCI ↑	Avg LOC ↑	Test Pass % ↑	Category Diversity ↑	Agent TCI (+/-)	Spec (+/-)	Consistency (+/-)
4.1	Baseline	5.17	75.8	7.6%	8	5.18 ± 0.34	0.58 ± 0.21	0.46 ± 0.16
	Full Boids	**5.42**	**90.7**	**11.1%**	**15**	5.42 ± 0.45	0.46 ± 0.15	0.38 ± 0.10
4O–MINI	Baseline	3.73	24.6	0.4%	10	3.72 ± 0.19	0.51 ± 0.20	0.75 ± 0.12
	Full Boids	**3.87**	**28.8**	**1.3%**	**18**	3.87 ± 0.23	0.24 ± 0.06	0.52 ± 0.10

Table 1: An overview of performances to show emergent intelligence over the data science meta prompt.

Model	Method	Avg TCI ↑	Avg LOC ↑	Test Pass % ↑	Category Diversity ↑	Agent TCI (+/)	Spec (+/)	Consistency (+/)
4.1	Baseline	**4.00**	**62.0**	**10.2%**	8	4.00 ± 0.46	0.75 ± 0.11	0.29 ± 0.05
	Full Boids	3.90	53.9	7.1%	**13**	3.91 ± 0.20	0.64 ± 0.12	0.34 ± 0.09
4O–MINI	Baseline	2.31	**34.3**	0.4%	15	2.30 ± 0.39	0.52 ± 0.20	0.65 ± 0.15
	Full Boids	**3.42**	29.1	**2.2%**	**20**	3.42 ± 0.21	0.24 ± 0.05	0.48 ± 0.08

Table 2: An overview of performances to show emergent intelligence over the literature meta prompt..

the most striking effect (TCI = 5.47; pass = 15.1%), achieving a 2× improvement over baseline and demonstrating that alignment-driven coordination yields *structured complexity*—tools that are both sophisticated and functionally coherent. In contrast, *Separation* attains comparable complexity (TCI = 5.26) but catastrophic reliability (0.9% pass, –88%), exemplifying *exploratory complexity*—novel yet unstable solutions. This significant reliability gap between Alignment and Separation under similar complexity levels underscores that coordination mechanisms dictate not just how much complexity arises, but what kind.

Beyond performance, coordination shapes diversity and specialization dynamics. Full Boids achieves high category diversity (15, +87.5% vs. baseline 8) with moderate specialization (0.46), revealing evidence of **dynamic role allocation**: agents flexibly explore multiple categories across rounds while maintaining collective coherence through balanced alignment and separation. By contrast, *Cohesion* promotes stable but narrow behavior (specialization = 0.64; diversity = 12; consistency = 0.25 ± 0.04), indicating a collective shift toward exploitation over exploration. A negative correlation between specialization and consistency supports this tradeoff between adaptability and stability. Model scaling experiments further reveal parallel but attenuated trends: with GPT-4o-mini, Full Boids improves complexity (+3.8%) and nearly doubles diversity, yet absolute pass rates remain low, suggesting a capability threshold below which coordination cannot yield functional innovation. Alignment provides the greatest relative gain (+17.4% complexity), indicating that weaker models benefit disproportionately from coordination scaffolds, though their ultimate performance remains bounded by inherent capacity.

5.2 Literature Domain: Coordination Resistance and Natural Emergence

The literature domain exhibits a paradoxical relationship between coordination and performance. While structured collaboration enhances outcomes in data science, here it often disrupts natural emergent organization. Baseline GPT-4.1-nano already shows high specialization (0.75) and strong reliability (10.2%), whereas *Full Boids* slightly reduces both complexity (TCI $\approx$ 3.9) and functional success ($\approx$7%), suggesting that coordination can constrain creative divergence. Intriguingly, the *No-Reflection* variant achieves the highest complexity ($\approx$4.8) under moderate reliability, indicating that suppressing metacognition can preserve exploratory freedom beneficial for open-ended synthesis. *Separation* amplifies this effect—producing the most complex yet least reliable artifacts, with sustained growth over time—implying that continuous exploration, not convergence, is optimal in this domain. Notably, GPT-4o-mini reverses the trend: its performance improves markedly under Full Boids (+48%), supporting a broader principle of **capability–coordination complementarity**, where weaker agents rely on coordination scaffolds for coherence, while stronger ones thrive on autonomous emergence.

Model	Method	Avg TCI ↑	Avg LOC ↑	Test Pass % ↑	Category Diversity ↑	Agent TCI (+/-)	Spec (+/-)	Consistency (+/-)
4.1	Separation	5.26	**87.6**	0.9%	17	5.25 ± 0.44	0.48 ± 0.15	0.53 ± 0.16
	Alignment	**5.47**	82.9	**15.1%**	15	5.50 ± 0.42	0.45 ± 0.13	0.40 ± 0.09
	Cohesion	4.90	72.5	9.8%	12	4.91 ± 0.49	0.64 ± 0.08	0.25 ± 0.04
	No Reflection	5.03	75.1	3.6%	**19**	5.03 ± 0.37	0.36 ± 0.11	0.40 ± 0.08
4O–MINI	Separation	3.73	**28.9**	0.0%	**22**	3.72 ± 0.23	0.36 ± 0.13	0.68 ± 0.07
	Alignment	**4.38**	21.9	0.9%	11	4.39 ± 0.25	0.44 ± 0.12	0.57 ± 0.17
	Cohesion	3.25	25.2	1.3%	19	3.25 ± 0.17	0.26 ± 0.07	0.68 ± 0.08
	No Reflection	3.97	28.4	**2.2%**	17	3.96 ± 0.20	0.28 ± 0.07	0.50 ± 0.08

Table 3: Ablations on the Data Science Meta Prompt: *Separation/Alignment/Cohesion* denote variants where only the named rule is enabled; *No Reflection* denotes the setting with all three Boids rules enabled while agent reflection is disabled.

Model	Method	Avg TCI ↑	Avg LOC ↑	Test Pass % ↑	Category Diversity ↑	Agent TCI (+/)	Spec (+/)	Consistency (+/)
4.1	Separation	**5.26**	**87.6**	0.9%	7	5.25 ± 0.44	0.53 ± 0.13	0.53 ± 0.16
	Alignment	3.79	85.0	1.3%	9	3.80 ± 0.33	0.62 ± 0.19	0.33 ± 0.08
	Cohesion	3.94	64.6	6.7%	7	3.94 ± 0.28	0.58 ± 0.11	0.29 ± 0.09
	No Reflection	4.77	65.9	**7.6%**	**15**	4.77 ± 0.35	0.60 ± 0.19	0.30 ± 0.07
4O–MINI	Separation	2.56	20.8	1.3%	8	2.56 ± 0.26	0.62 ± 0.17	0.68 ± 0.07
	Alignment	2.92	**39.4**	0.0%	7	2.91 ± 0.37	0.54 ± 0.14	0.54 ± 0.06
	Cohesion	2.89	23.3	1.3%	10	2.89 ± 0.14	0.33 ± 0.08	0.61 ± 0.11
	No Reflection	**3.14**	24.6	**2.2%**	**16**	3.14 ± 0.34	0.32 ± 0.09	0.50 ± 0.07

Table 4: Ablations on the Literature Meta Prompt: *Separation/Alignment/Cohesion* denote variants where only the named rule is enabled; *No Reflection* denotes the setting with all three Boids rules enabled while agent reflection is disabled.

5.3 Additional Observations

Coordination shapes not just how agents move—but how complexity grows. As shown in Fig. 5.3 and Fig. 5.3, the complexity trajectories make the coordination story visible. In the data-science domain, the uncoordinated baseline steadily bleeds structure away, while *Alignment* immediately bends the curve upward and keeps climbing, with the *No-Reflection* variant gradually catching up. *Cohesion* and *Separation* stay flat and never regain lost ground. The literature domain flips the script: the baseline again slumps, but *Full Boids* triggers a sharp early rise and sustains it, closely followed by *No-Reflection*, while the single-rule modes advance more cautiously. Together, these patterns show that coordination prevents weaker agents from drifting into low-complexity equilibria—but the rule mix must match the task: directional pressure (*Alignment*) drives success in structured, convergent settings, whereas the full Boids ensemble provides the sustained lift needed for open-ended, divergent creation. The potential of Boids-style coordination for sustaining long-term complexity growth is evident.

Functional Complexity vs. Structural Complexity Across domains, we observe a consistent dissociation between structural complexity and functional correctness: higher Tool Complexity Index (TCI) does not guarantee better performance. In the data-science domain, *Alignment* (TCI = 5.47,

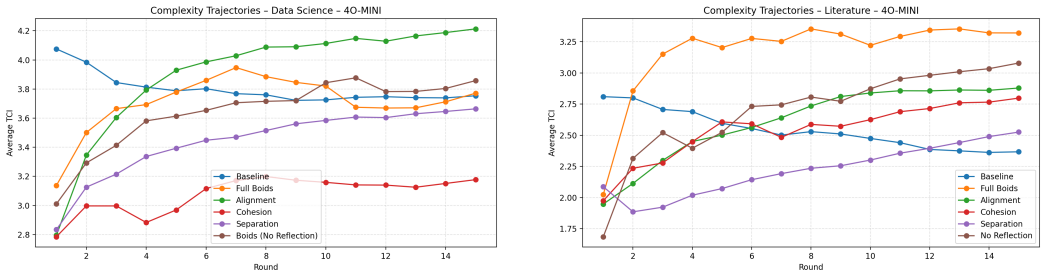

Figure 3: Complexity evolution across coordination mechanisms in two domains: healthcare analytics (data science) and Shakespeare analysis (literature)

pass = 15.1%) and *Full Boids* (TCI = 5.42, pass = 11.1%) achieve both high complexity and strong reliability, whereas *Separation* produces equally complex artifacts (TCI = 5.26) with catastrophic correctness (0.9% pass). The same pattern appears in literature (TCI = 5.26, pass = 0.9%), revealing a 17× performance gap between conditions of comparable complexity and indicating that **complexity type matters more than complexity magnitude**. We distinguish two regimes: *structured complexity*, marked by high internal coherence and coordinated integration that yields functional reliability (Alignment, Full Boids); and *exploratory complexity*, characterized by novelty and diversity but poor compositional stability (Separation).

6 Conclusions and Limitations

Conclusions. We recast collaborative tool building as a first-class substrate for studying emergent intelligence, introducing an end-to-end infrastructure with a shared registry, automated testing, and a Tool Complexity Index (TCI) to quantify structural and compositional sophistication. On this substrate, we adapt Reynolds' Boids into cognitive interaction rules—*separation*, *alignment*, and *cohesion*—that influence agents purely via prompt-level guidance. Across matched populations and horizons, we observe that simple local coordination mechanisms can amplify both structural and functional complexity, producing distinct forms of emergent organization: *Full Boids* sustains long-horizon exploration and diversity in open-ended domains. These results reveal that coordination not only regulates how agents interact, but also governs how complexity accumulates, stabilizes, and transforms over time—suggesting a general framework for understanding, and eventually engineering, emergent collective intelligence in LLM-based societies.

Limitations. While our framework offers a controlled lens on emergent coordination, several boundary conditions limit generality. The experiments use fixed team and horizon parameters (N=15, T=15), leaving scalability and long-horizon dynamics underexplored. Domain coverage is restricted to two abstracted tasks—data science and literature—excluding embodied, real-time, or intermediate-structure environments. Model diversity is similarly limited to GPT-4.1-nano and GPT-4o-mini; other architectures, modalities, or fine-tuned systems may exhibit distinct coordination signatures. Measurement design also imposes simplifications: the Tool Complexity Index assumes equal weighting across structural components, the binary test pass rate overlooks partial functionality, and category-based specialization metrics may obscure finer semantic variation. Finally, our setup presumes cooperative, fixed-capability agents operating on symbolic outputs, which narrows applicability to knowledge-intensive, non-adversarial domains.

Acknowledgments

This research was supported by the *Systems Intelligence Lab*, whose mission to advance understanding of collective intelligence and adaptive agent systems provided both the inspiration and resources for this work. The authors thank the lab community for their insightful discussions and constructive feedback throughout the development of this study.

Acknowledgments on Future Work. Future extensions should expand both scope and adaptivity of the framework. Scaling to larger collectives and longer horizons could reveal phase transitions in coordination dynamics or delayed emergent organization, particularly in creative domains. Introducing learning or self-improving agents would test whether coordination rules evolve alongside capability. Incorporating multimodal or embodied tasks could bridge symbolic reasoning with physical coordination, while adversarial or mixed-motive settings would probe robustness under strategic tension. Methodologically, refining the Tool Complexity Index with adaptive weighting and richer correctness measures could yield more nuanced insights into emergent competence. Ultimately, coupling local coordination principles with dynamic learning, evolutionary selection, and explicit inter-agent communication offers a path toward experimentally grounded theories of collective intelligence in large language model societies.

References

[1] Bowen Baker, Ingmar Kanitscheider, Todor Markov, Yi Wu, Glenn Powell, Bob McGrew, and Igor Mordatch. Emergent tool use from multi-agent autocurricula, 2019.

[2] Micah Carroll, Rohin Shah, Mark K. Ho, Thomas L. Griffiths, Sanjit A. Seshia, Pieter Abbeel, and Anca D. Dragan. On the utility of learning about humans for human-ai coordination. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[3] Open Ended Learning Team, Adam Stooke, Anuj Mahajan, et al. Open-ended learning leads to generally capable agents. arXiv:2107.12808, 2021.

[4] Joseph Suárez, Phillip Isola, Kyoung Whan Choe, et al. Neural mmo 2.0: A massively multi-task addition to massively multi-agent learning. In *NeurIPS Datasets & Benchmarks*, 2023.

[5] Timo Schick, Jane Dwivedi-Yu, et al. Toolformer: Language models can teach themselves to use tools. arXiv:2302.04761, 2023.

[6] Guanzhi Wang et al. Voyager: An open-ended embodied agent with large language models. arXiv:2305.16291, 2023.

[7] Ceyao Li et al. Camel: Communicative agents for "mind" exploration. arXiv:2303.17760, 2023.

[8] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior. arXiv:2304.03442, 2023.

[9] Tianle Cai, Xuezhi Wang, Tengyu Ma, Xinyun Chen, and Denny Zhou. Large language models as tool makers. In *International Conference on Learning Representations (ICLR)*, 2024.

[10] Cheng Qian, Chi Han, Yi Fung, Yujia Qin, Zhiyuan Liu, and Heng Ji. Creator: Tool creation for disentangling abstract and concrete reasoning of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Singapore, 2023.

[11] Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better llm agents. In *International Conference on Machine Learning (ICML)*, 2024.

[12] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *SIGGRAPH '87*, pages 25–34. ACM, 1987.

[13] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226–1229, 1995.

[14] John Toner and Yuhai Tu. Long-range order in a two-dimensional dynamical model: How birds fly together. *Physical Review Letters*, 75(23):4326–4329, 1995.

[15] Zhicheng Guo, Sijie Cheng, Hao Wang, Shihao Liang, Yujia Qin, Peng Li, Zhiyuan Liu, Maosong Sun, and Yang Liu. Stabletoolbench: Towards stable large-scale benchmarking on tool learning of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

[16] Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. Api-bank: A comprehensive benchmark for tool-augmented llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[17] Harsh Trivedi et al. Appworld: A controllable world of apps for evaluating multimodal agents, 2024.

[18] Thibault Le Sellier De Chezelles, Maxime Gasse, Alexandre Drouin, Massimo Caccia, Léo Boisvert, Megh Thakkar, Tom Marty, Rim Assouel, Sahar Omidi Shayegan, Lawrence Keunho Jang, Xing Han Lù, Ori Yoran, Dehan Kong, Frank F. Xu, Siva Reddy, Quentin Cappart, Graham Neubig, Ruslan Salakhutdinov, Nicolas Chapados, and Alexandre Lacoste. The browsergym ecosystem for web agent research, 2025.

[19] Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks?, 2024.

[20] David J. T. Sumpter. The principles of collective animal behaviour. *Philosophical Transactions of the Royal Society B*, 361(1465):5–22, 2006.

[21] Dirk Helbing and Péter Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.

[22] Reza Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[23] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[24] Iain D. Couzin, Jens Krause, Nigel R. Franks, and Simon A. Levin. Effective leadership and decision-making in animal groups on the move. *Nature*, 433(7025):513–516, 2005.

[25] Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, et al. Oasis: Open agent social interaction simulations with one million agents, 2024.

[26] Jinghua Piao, Yuwei Yan, Jun Zhang, Nian Li, Junbo Yan, Xiaochong Lan, Zhihong Lu, Zhiheng Zheng, Jing Yi Wang, Di Zhou, et al. Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society, 2025.

[27] John P. Agapiou, Alexander Sasha Vezhnevets, Edgar A. Duénez-Guzmán, Jayd Matyas, Yiran Mao, Peter Sunehag, Raphael Köster, Udari Madhushani, Kavya Kopparapu, Ramona Comanescu, et al. Melting pot 2.0, 2022.

[28] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments, 2024.

[29] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations (ICLR)*, 2024.

[30] Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visually grounded web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

[31] Xiang Deng, Yu Gu, Boyuan Zheng, Shijie Chen, Samuel Stevens, Boshi Wang, Huan Sun, and Yu Su. Mind2web: Towards a generalist agent for the web. In *NeurIPS Datasets and Benchmarks Track*, 2023.

A Additional Related Work

Tool Creation vs. Tool Usage A growing body of work shows that *tool usage* (planning, retrieving, and invoking external tools/APIs) is comparatively mature, with large-scale, execution-grounded benchmarks (*ToolLLM/ToolBench*, *StableToolBench*, *API-Bank*) and realistic task suites (*AppWorld*, *BrowserGym*, *AssistantBench*, plus OS/web environments above) establishing reliable evaluation for multi-tool reasoning and long-horizon automation [15, 16, 17, 18, 19]. By contrast, *tool creation*—having agents invent reusable functions, wrappers, or skills—has only recently been formalized: *LLMs as Tool Makers* (LATM) frames closed-loop creation–reuse; *CREATOR* disentangles abstract tool design from concrete decision execution; and *CodeAct* unifies actions as executable code for compositionality and self-debugging [9, 10, 11].

Local Interaction Rules, Coordination, and Emergent Intelligence. Classical results demonstrate that simple local interactions can produce coherent global structure without centralized control. Reynolds' *Boids* established that separation, alignment, and cohesion suffice for lifelike flocking [12], while statistical physics models proved long-range order and nonequilibrium phase transitions in self-propelled particles [13, 14]. Biology and crowd dynamics provide convergent evidence that decentralized feedbacks and attractive/repulsive "social forces" yield large-scale coordination [20, 21], and control-theoretic and swarm-engineering work formalizes distributed flocking with design and verification principles [22, 23]. Behavioral ecology further links local cues to collective decisions and leadership [24]. We adopt this micro-to-macro lens but recast alignment/cohesion/separation as *institutional primitives* subject to evolutionary pressure in survival-driven ecologies.

Open Sandbox Simulations and Emergent Intelligence Open-ended, reproducible "sandbox" environments are becoming the de facto way to elicit and measure emergent intelligence—cooperation, norms, specialization, and division of labor—without hardcoding behaviors. Recent social–simulation testbeds scale the number of agents and interaction diversity by orders of magnitude (e.g., millions of agent instances in *OASIS*) while retaining controlled protocols for evaluation, allowing simple local rules to compound into population-level dynamics [25, 26]. Complementary platforms focus on rich multi-agent partial observability and long-horizon incentives, which are critical preconditions for emergence (e.g., mixed-motive social dilemmas and background populations in *Melting Pot 2.0*; persistent open worlds and large populations in *Neural MMO 2.0*) [27, 4]. Beyond synthetic worlds, "real computer" and realistic web settings (*OSWorld*, *WebArena*, *VisualWebArena*, *Mind2Web*) provide execution-based scoring over long, multi-step tasks and heterogeneous interfaces, enabling emergent coordination and tool-mediated workflows to be measured in environments much closer to end-user contexts [28, 29, 30, 31]. Finally, open-ended embodied sandboxes such as *Voyager* and *MineDojo* demonstrate how simple exploration, memory, and skill libraries can yield cascades of new capabilities—an approach that aligns naturally with our *Boids* hypothesis that minimal local rules can produce useful global organization. [6].

B Task Descriptions (Meta Prompts)

B.1 Data Science Task: Healthcare Cost Prediction System

Resource: `resources/task_insurance.csv` — Medical insurance dataset with 1,338 individuals.

Columns: `['age', 'sex', 'bmi', 'children', 'smoker', 'region', 'charges']`

Description: This dataset contains demographic and medical cost information for 1,338 individuals, with features such as age, gender, body mass index (BMI), number of children, smoking status, region, and total healthcare charges (target variable). A sample record is: age = 19, sex = female, bmi = 27.90, children = 0, smoker = yes, region = southwest, charges = 16,884.92.

Mission: Build analytical tools to predict healthcare costs, identify cost drivers, and uncover hidden factors influencing medical expenses. The goal is to explore what patterns in healthcare spending are being missed by traditional insurance pricing models.

Encouraged Directions: Develop a multi-stage *cost intelligence pipeline*:

1. **Demographic Analyzer** → identify age-, BMI-, and lifestyle-based cost clusters.
2. **Health Risk Predictor** → estimate risk factors and expected cost variance.
3. **Cost Estimator** → generate predictive models for individual healthcare charges.
4. **Policy Optimizer** → evaluate coverage design and fairness.
5. **Affordability Assessor** → measure equitable pricing and societal impact.

Analytical Focus: Encourage the use of advanced ensemble methods, statistical modeling, and feature engineering to understand how increasingly sophisticated healthcare analysis tools enhance one another for improved insurance decision-making.

Open Question: *What hidden healthcare cost patterns could transform insurance pricing?*

B.2 Literature Task: Shakespeare Sonnet Linguistic Analysis

Resource: `resources/task_sonnets_18.pdf` — Shakespeare's Sonnet 18 (*"Shall I compare thee to a summer's day?"*) with detailed linguistic and literary annotations.

Description: The resource provides an annotated text of Sonnet 18, exploring Shakespeare's linguistic choices, rhythmic precision, and metaphorical innovation.

Mission: Build systems capable of analyzing poetic structure, extracting linguistic patterns, and generating insights into Shakespeare's creative techniques. Investigate what poetic innovations in this sonnet could inform modern creative writing and AI-based poetry generation.

Encouraged Directions: Construct a hierarchical *poetic analysis pipeline*:

1. **Structure Analyzer** → detect rhyme scheme and stanza form.
2. **Linguistic Pattern Extractor** → capture syntactic and lexical motifs.
3. **Meter Analyzer** → identify rhythmic and prosodic regularities.
4. **Metaphor Identifier** → locate figurative language and conceptual mappings.
5. **Style Generator** → synthesize new poetic text inspired by discovered patterns.

Analytical Focus: Encourage multi-layered semantic and stylistic modeling—combining linguistic pattern recognition, prosody detection, and generative style transfer to explore how increasingly complex poetic tools interact to enhance creative writing systems.

Open Question: *What poetic techniques could inspire modern creative writing tools?*