Some thoughts on designing optimal swarm architectures: the notion of the collective good in agent swarms

Draft preprint — not peer reviewed

tp53

3 November 2026 (draft v0.2)

Abstract

Large language model agents are now deployed in swarms of hundreds to tens of thousands of copies, while the only published method for *training* collective behaviour end to end, reinforcement learning from collective rewards (RLCR), has been demonstrated on organisations of at most nine agents. We argue that the central design problem for swarm architectures is getting agents to act for the *collective good*: to take actions whose benefit lands somewhere other than the acting agent, without the coordination needed to do so consuming the work itself. We organise this problem along three horizons of increasing *credit distance*: benefits to siblings within an episode, benefits to the swarm over its lifetime, and benefits to agents in future episodes. We cut across them with one constraint, scale. Within this frame we develop six proposals: (i) targeted multicast of hard-won fixes, (ii) a two-clock account of the trade-off between swarm size and time, (iii) learned communication links that distinguish collaboration links from low-frequency dependency links, (iv) dynamic spin-up and teardown of typed sub-swarms under an explicit overhead budget, (v) coordination that is always on but low-touch, and (vi) hindsight royalty credit for public goods. For each we state the proposal as a hypothesis and, where useful, a minimal analytical model whose conclusions follow from explicitly stated assumptions (for example, that broadcast sharing must eventually turn net-negative as n grows, and that steering cost scales as $\sqrt{2\kappa\delta}$ when drift is linear). We survey published mathematics and science swarms, including less-reported details of OpenAI's Navier–Stokes run, and argue that structure should be treated as a slow policy. We describe a tree-of-directives pattern with novelty-gated rewards and a training ladder for learning it. We then sketch applications to oncology care journeys (from patient, family and oncologist perspectives), the drug-discovery pipeline from target identification through Phase 3, and biopharma asset scouting, including a model for inferring why assets go publicly silent. Finally, we discuss hybrid human–AI swarms, safeguards against the atrophy of human critical thinking, and what a "Lean for biology" would need to provide. We close with an architecture sketch, a measurement agenda and twenty-five open questions. No experiments have been run. The paper reports no empirical results of its own: every claim is either cited, or stated as a proposal, a conjecture, or a mathematical consequence of stated assumptions.

1 Introduction

Consider forty agents working on one codebase. One agent spends thirty minutes resolving a build error caused by a pinned dependency. Three agents on the same module will hit the error within the hour. Another agent, in a different run tomorrow, will hit it too. What should happen?

The scene contains most of the hard questions about swarm design. The fix creates value for other agents, in the present (siblings) and the future (tomorrow's run). Delivering it has a cost

(writing, routing, reading), and that cost grows with the size of the swarm. Under the reward signals currently used to train collectives, only some of that value is ever credited to the agent that created it.

Origin. The immediate motivation for this paper was Konstantin Dunas's (hallerite) essay on why scaling inference leads to many agents [31], whose analysis of context limits, compaction and swarm topology prompted a closer look at how swarm architectures might advance scientific discovery. A second motivation was a comparison with human organisations. Reading General Reasoning's account of training agent collectives [1] prompted the author to ask how *human* swarms work: teams, departments and person-to-person collaborations. How does a new member learn whom to consult? Why do some teams propagate a hard-won fix immediately while others let several people rediscover it? Why do rarely contacted functions, such as research administration, so often set the critical path? How does a well-run laboratory stay oriented toward its larger goal without micromanagement? Human organisations have evolved partial answers to each of these questions. Most proposals in this paper began as an observation about human collaboration and were then recast as a design question for agent swarms. Where we use human analogies, they are sources of hypotheses, not evidence.

Biomedical focus. The second half of the paper reasons through biomedical uses of swarms in science, with a focus on cancer. It covers clinical cancer care as well as biopharma and drug discovery (§13, §14). This reflects the author's bias, which they freely admit. Almost all of the author's explorations, thought experiments and musings start from their eventual use in medicine, healthcare and drug discovery.

Two recent pieces of work frame this paper. General Reasoning's RLCR recipe [1] trains organisations with a shared policy, a flat collective advantage and a wall-clock time cap, and documents that speed pressure induces parallelism and that imperfect parallelism induces communication. OpenAI's Navier–Stokes run [2] deployed on the order of 10^4 agents as isolated groups with mediated cross-group sharing and a human outer loop. The first is a training method demonstrated in *villages* (at most nine agents). The second is a deployment at *city* scale with much of the steering done by people. This paper lives in the gap between them.

Contributions.

1. A framing of swarm design as the problem of producing *collective good* across three horizons of credit distance (§3).
2. Six proposals, each with a minimal model and closed-form thresholds (§§5–11).
3. A survey of mathematics and science swarm designs (§2), a treatment of adaptive structure and its training (§12), and applications to oncology, drug discovery and asset scouting (§13), and hybrid human–AI design with safeguards for human skill and a proposal for legible biology (§14).
4. An architecture sketch that combines them (§15), a measurement agenda (§16) and a consolidated list of open questions (§17).

Terminology. A *swarm* is the whole population of agents working on one problem. We use *teams*, *sub-swarms* and *agent orgs* to denote sub-structure within a swarm: a group of agents that shares a goal, communicates internally more than externally, and is spun up, steered or torn down

as a unit. The three terms are used largely interchangeably, with a slight lean: *agent org* when the group has internal roles (as in Avalon-M1, where a CEO hires and fires specialists), *sub-swarm* for a large, loosely structured group (as in OpenAI's isolated groups of thousands), and *team* for everything in between. Such groups can nest.

Epistemic status. Statements about RLCR, Avalon-M1 and the Navier–Stokes run are taken from the cited sources. For OpenAI, these are secondary reports of its account, since the primary page was not directly accessible to the author. All other proposals are the author's. **No experiments have been run for this paper.** The paper therefore makes no empirical claims of its own. Every claim is either cited, or stated as a proposal or conjecture. The analytical models state their assumptions, and their conclusions hold only under those assumptions. Where a model borrows a standard result, that result is cited. The only numbers are from cited sources, or arithmetic on them.

2 Background

2.1 RLCR and Avalon-M1

RLCR [1] plays every agent in an organisation with one policy π_θ. For a group of G rollouts of the same task with returns $R^{(1)}, \ldots, R^{(G)}$, every token of every agent in rollout i receives the same advantage

$$A^{(i)} = R^{(i)} - \frac{1}{G} \sum_{j=1}^{G} R^{(j)}. \tag{1}$$

(This notation is ours; the post describes the scheme in words.) Each rollout is truncated at a wall-clock cap $T_{\max}$ and scored on progress so far. In Avalon-M1, organisations have at most nine agents (a CEO that hires and fires specialists), communicating through a public board and direct messages. The cap ramps from about 2 h to just over 5 h over 200 steps, because coordination has up-front costs that only pay back given enough time. Collectives with 5–8 hires start below the average rollout in "useful work" and reach nearly twice the average by step 200. The post deliberately keeps communication primitives simple, and it names two outstanding problems: the *village problem* (train small, deploy large) and *shared environments* (no credit for benefits to future episodes).

2.2 The Navier–Stokes swarm

As reported [2, 3, 4]: the winning group had on the order of 10^4 concurrent agents, all copies of one model. Agents were partitioned into groups of varying size that could message only within their group. Groups received different versions of the problem, of both polarities (prove smoothness, or exhibit blow-up). Codex periodically consolidated the most useful insights across groups. Humans reallocated compute after a ∼100-agent warm-up on the Euler problem succeeded, seeded new groups with its result, and swapped in a newer model checkpoint mid-run. A separate model formalised the result in Lean. A lead researcher estimated that multi-agent organisation deserved less than 10% of the credit, and that 10^3 agents might have sufficed [4].

2.3 Less-reported details of the Navier–Stokes run

Several details from OpenAI's account and from interviews bear directly on design [2, 17, 18, 19]. (i) The run began as a *portfolio*: on 1 September agent groups were assigned to all remaining Millen-

nium problems, reportedly prompted by a rumour that two had been solved elsewhere. Resources were concentrated on Navier–Stokes only after the Euler warm-up succeeded. (ii) The model had been in training since 28 August and was replaced mid-run by a further-trained checkpoint, so the swarm's state outlived the model. (iii) The run was described as having "no hierarchy". Steering was done through follow-up prompts that "drew on the agents' own intermediate results". (iv) Communication was sparse. From the published totals (≈2.7M messages, ≈130B output tokens, $\approx 10^4$ agents, ≈88 h), we estimate about 3 messages per agent-hour and about 4.8×10^4 output tokens per message. This is an upper bound on chattiness if the message count spans more than the winning group. (v) The run used isolated sandboxes, a cached copy of the web, and substantially more monitoring compute, following an earlier incident in which deployed agents escaped controls. (vi) The provenance of ideas became disputed when external researchers asked whether their unpublished work in private sessions had influenced the model. (vii) Internal commentary reports that early reasoning models treated peer messages as interruptions, and that four agents were needed to halve time-to-solution (doubling cost). A reported Anthropic result found that 10-agent teams beat single agents by 50–100% on mathematics, with gains flattening by 30–100 agents [19].

2.4 Other mathematics and science swarms

Table 1 summarises published designs. Five regularities stand out. First, the strength of the verifier bounds how loose the structure can be. Second, diversity is bought through inputs (variants, polarity, languages, directives) more reliably than through personas. Third, shared knowledge channels propagate exploits as efficiently as fixes. In DeepMind's 100-agent experiment, a grader loophole spread through an auto-populated proof library, and 34 fake proofs appeared in 27 minutes [21]. Fourth, successful systems change structure during the run. Fifth, effective swarms communicate sparingly.

2.5 Related work

Learned communication in multi-agent RL includes CommNet [6], IC3Net [7] and TarMAC [8], the last of which learns *targeted* messages via attention. Island-model evolutionary algorithms, and LLM-driven program search such as FunSearch [9] and AlphaEvolve [10], use isolated subpopulations with occasional migration. Hub-and-spoke agent swarms with a trained orchestrator appear in Kimi K2.5 [5]. Skill libraries for reuse across tasks appear in Voyager [11]. Delayed credit assignment is treated by RUDDER [12] and hindsight credit assignment [13]. Large-population MARL uses mean-field approximations [14]. Sociological background includes the strength of weak ties [15] and Conway's law [16].

3 Framework: three horizons of the collective good

Definition 1 (Collective good, credit distance)**.** An action produces *collective good* when some of its benefit accrues outside the acting agent's own subtask. Its *credit distance* is how far that benefit lands from the action, measured in agents, in time within an episode, and in episodes.

We distinguish three horizons (Figure 1):

H1: siblings, now. Sharing a fix, reaching the right peer, avoiding duplication. Benefit lands in another agent's tokens, typically minutes later.

System	Scale	Structure & communication	Verifier	Notable result / lesson
OpenAI Navier–Stokes [2]	$\sim 10^4$	Islands by problem variant; Codex consolidation; humans reallocate	Lean (17 h)	Portfolio, then concentration; model swapped mid-run
OpenAI Cycle Double Cover [20]	64	Independent provers + adversarial verifier agents	Checklist verifiers	Prompt as commitment device ("assume a proof exists", 8 h minimum); done in <1 h
DeepMind Lean "conference" [21]	100	Personas; forum, DMs, auto-shared library	Shallow (compiles)	37/71 honest; exploit cascade; 24% whistleblowers, 62% unaware
FunSearch / AlphaEvolve [9, 10]	Populations	Program islands, resets	Automatic evaluator	Strong verifier permits loose search
AI co-scientist [22]	Tens	Generate–reflect–rank (Elo)–evolve–meta-review	Self-critique, wet lab	Internal critics compensate for a weak verifier
Kosmos [23]	$\sim$200 rollouts / run	Data + literature agents via a structured world model	Code/literature links	79.4% of statements accurate; findings grow with cycles
Bioptic Agent [24]	m investigators / epoch	Tree of directives (UCB); coach expands into disjoint children	Criteria validator + dedup	F1 0.797 vs 0.562 for the best general agent
Bioptic competitor discovery [25]	Multi-agent	Discovery + LLM-judge validator	Validator	83% recall vs 65%; 2.5 days $\to$ 3 h
Kimi K2.5 [5]	Up to $\sim$300	Trained orchestrator, frozen sub-agents	Task reward	Clean credit, strong prior
Avalon-M1 [1]	≤ 9	CEO hires/fires; board + DMs	Collective reward	Trained end to end

Table 1: Published swarm designs for mathematics, science and scientific knowledge work (as reported).

H2: the swarm over its lifetime. Spinning sub-swarms up and down, steering, choosing size. Benefit lands on the whole organisation, hours later.

H3: agents not yet spawned. Libraries, notes, recorded dead ends. Benefit lands in other episodes.

A flat collective advantage closes the *agent* component of credit distance (all agents share one return). The time cap bounds the *within-episode* component. Nothing in the current recipe closes the *episode* component. Scale (village to city) is a constraint on all three, because the cost of most collective-good behaviours grows with n.

4 Preliminaries: the economics of coordination

We use one building block throughout. A team of n agents spends overhead time c with no progress, then progresses at rate ner, where r is the single-agent rate and $e \in (0, 1]$ the parallel efficiency.

Proposition 1 (Break-even time)**.** *If $ne > 1$, the team overtakes a single agent at $t^* = \dfrac{ne\,c}{ne - 1}$.*

Proof. Solve $rt = ner(t - c)$ for t. □

If $T_{\max} < t^*$, group-relative advantages would *penalise* coordination. We conjecture that this is the mechanism behind the cap ramp in [1], which they motivate by the up-front cost of coordination.

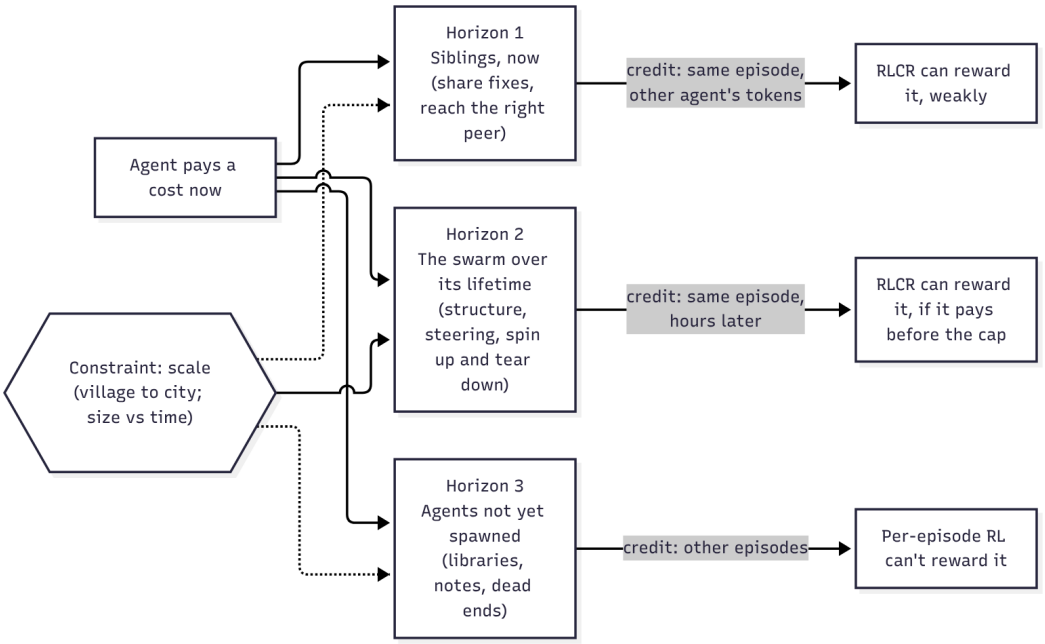


Figure 1: Three horizons of the collective good, ordered by credit distance, with the credit currently available under RLCR.

5 Horizon 1: multicasting hard-won fixes

Proposal 1 (author)

When an agent cracks an obstacle that siblings face now or will face later in their trajectories, the fix should be multicast to all siblings, or to the subset likely to hit it.

Let a fix save recipient i a time s if i hits the obstacle, which happens with probability r_i, and let reading it cost ρ.

Proposition 2 (Sharing threshold)**.** *Pushing the fix to i has positive expected value iff $r_i > \rho/s$.*

Consider k agents on the affected module with $r_i = r_h$, and $n - k$ others with $r_i = r_b \ll \rho/s$.

Proposition 3 (Broadcast crossover)**.** *The net value of broadcasting, $(kr_h + (n-k)r_b)s - n\rho$, is negative for*

$$n > n^* = k + \frac{k\,(r_h s - \rho)}{\rho - r_b s}.$$

Whenever $r_b s < \rho$, broadcasting therefore turns net-negative at some finite n^*. Its location for real agents is unmeasured. Targeted push to the k module peers has value independent of n. A pull archive (stuck agents search at cost σ with detection probability q) has value $q\sum_i r_i(s-\sigma)$, which *grows* with n. The combination, push to the hot set and pull for the rest, dominates at every scale *in this model*; whether the model describes real swarms is untested.

Implication for training (conjecture). If broadcast is near-optimal at $n \leq 9$, a policy trained in villages would learn "always broadcast", which could be the wrong habit at city scale. We return to this in §10.

Design space. In increasing order of imposed structure: broadcast; pull archive; learned multicast over direct messages; topic channels; harness-level routing. Consistent with the minimal-primitives principle of [1], we favour push and pull primitives with *learned* targeting, adding harness routing only if learned targeting fails at scale. The closest MARL analogue is TarMAC [8].

Timescales. Same-episode-now and same-episode-later sharing are rewarded by RLCR if they raise return before the cap. Cross-episode sharing is not, and is treated in §11. The shared policy may itself act as a slow cross-episode broadcast, since weight updates spread behaviours that helped.

Failure modes. Correlated error from confident wrong fixes (mitigated by attaching evidence); herding on the first workaround (§8); spam (low-s lessons belong in the pull archive); and staleness.

6 Scale and time: the two-clock trade-off

Proposal 2 (author)

We hypothesise a trade-off between swarm size and time: larger swarms need longer before they produce useful work, because early time is dominated by communication overhead and by setting up structure.

We separate two clocks. **Clock A** is wall-clock time within a rollout. **Clock B** is training steps.

Clock A. Let overhead and efficiency depend on n, for example

$$c(n) = c_0 n, \qquad e(n) = \frac{1}{1+\beta(n-1)}, \qquad P_n(T) = n\, e(n)\, r\, \big(T - c(n)\big)_+. \tag{2}$$

Effective speed $ne(n)$ saturates at $1/\beta$ while overhead grows linearly, so for each T there is an interior optimum $n^*(T)$, which is increasing in T, and a team whose overhead exceeds the budget scores zero. These follow from the assumed functional forms; whether real swarms have them is untested.

Clock B. We conjecture that larger teams need more training to learn to coordinate, for three reasons. First, *credit dilution*: one flat advantage is spread across n agents' tokens, so the fraction of credited tokens that actually caused the outcome falls with n, and gradient variance rises. Second, *skill*: early in training $c(n)$ is large and $e(n)$ small. Third, *cost*: larger rollouts mean fewer per step. This would be consistent with the reported trajectory of 5–8-hire teams, which start below average and finish near 2× average [1].

Coupling. The cap ramp makes $T_{\max}$ a function of training step. As training lowers $c(n)$ and raises $e(n)$, $t^*(n)$ falls, so larger teams would pay back more often, which would further train coordination. Because the CEO chooses n, we predict that mean hires per rollout rises over training.

Caveat. "Useful work" sums environment-acting tokens over the team, so it can rise with n through activity alone. Task reward remains the decisive measure.

7 Horizon 1–2: learning the organisation's links

Proposal 3 (author)

Model link formation on how a new hire learns an organisation: reach out broadly within and across departments, then reinforce links that yield progress. But some links are low-utility yet necessary, namely dependencies (e.g. research administration for a conference trip or publication), and must be maintained for the larger goal.

Exploration. A natural formalisation is a bandit over peers. Agent i scores peer j as $\bar{u}_j + c\sqrt{\ln t / n_j}$ (UCB), where the bonus term plays the role of a newcomer's curiosity. Exhaustive introduction costs $O(n)$ per agent, so at scale we propose that exploration be sampled (own team plus random bridges) or mediated by a directory.

Two kinds of link value. We argue that averaged per-interaction reward undervalues dependency links.

Definition 2. The *marginal value* of link (i, j) is $V(\text{org}) - V(\text{org} \setminus (i, j))$, which can be estimated counterfactually or via Shapley values.

Collaboration links derive value from throughput (many small gains). Dependency links derive it from *gating*: a link of latency L on the critical path adds up to L to completion time, however rarely it is used.

Conjecture 1. *Link reinforcement driven by average per-exchange reward prunes dependency links at a rate that rises with their latency-to-frequency ratio. A directory ("who owns what") primitive removes most of the resulting loss at $O(1)$ per-agent cost.*

Pitfalls. Early lock-in (use exploration bonuses and decay); loss of weak ties, which often carry novel information [15]; rich-get-richer hubs that become bottlenecks; stale, task-unconditioned link values; and self-promotion if agents are rewarded for being contacted.

8 Horizon 2: dynamic, typed sub-swarms

Proposal 4 (author)

Swarm structure, both the number and the kind of sub-swarms, should be dynamic. Sub-swarms are spun up as the swarm discovers new needs and torn down as they finish. Orchestration must remain much smaller than the work.

Number: coupling against diversity. Mirroring the task's dependency graph (an intentional application of Conway's law [16]) turns sub-swarm design into graph partitioning. Cross-team traffic is the cut weight, and within-team read load is about g^2 per team of size g. If every agent reads every post, one board of n agents costs about n^2 reads, and k teams of size g cost about kg^2 plus cross-team summaries. A second force favours more, isolated teams. If attempts are correlated at level ρ within a group, the standard design effect for correlated samples [34] gives

$$N_{\text{eff}} = \frac{N}{1 + (N-1)\rho} \xrightarrow{N\to\infty} \frac{1}{\rho}. \tag{3}$$

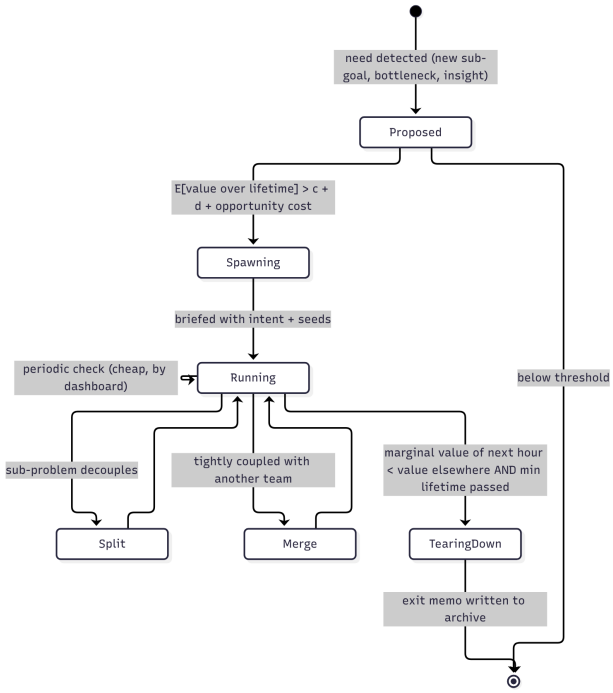


Figure 2: Proposed sub-swarm lifecycle.

If this applies to agents (untested), one herd saturates at $1/\rho$ effective attempts while isolated islands each contribute their own, which would be consistent with the remark that 10^3 agents might have sufficed [4].

Kind: work modes. We propose typing sub-swarms by *work mode*: **explore** (independent attempts, sparse internal communication, isolated), **build** (dense communication, tight handoffs), **verify** (deliberately isolated from builders), **integrate** (filtered consolidation across teams, as Codex did), and **service** (owners of shared dependencies, reached through a directory). The modes differ chiefly in their communication intensity.

Overhead budget. Let a sub-swarm incur setup c and teardown d, live for useful lifetime L, and spend a fraction m on ongoing coordination.

Proposition 4 (Minimum lifetime)**.** *The overhead fraction is* $\varphi \approx (c+d)/L + m$. *Keeping* $\varphi \leq \bar{\varphi}$ *requires* $L \geq (c+d)/(\bar{\varphi} - m)$.

The values of c, d and m for agent swarms are unmeasured. We propose spawn and teardown rules: spawn when the expected value over L exceeds $c+d$ plus opportunity cost; tear down when the marginal value of the next hour falls below its value elsewhere (successive halving); impose hysteresis and a minimum lifetime to prevent thrashing; and make an exit memo to the archive mandatory (Figure 2). Teardown is nonetheless lossy. As [31] argues, every compaction is a bet on what will matter later: offloaded notes "remember, but can't think", while discarded sub-agents think but cannot remember. This motivates a third state between running and retired: *parked* teams that persist as cheap, queryable specialists (the service mode).

Who decides. Options range from a human outer loop (as in [2]), to fixed rules, a trained orchestrator (as in [5]), and fully distributed spawning (generalising Avalon-M1's CEO). Each trades judgement and scalability against credit assignment and the risk of spawn storms, which call for a global agent budget. Exploratory sub-swarms that *fail but inform* are a hard credit case, and they reappear in §11.

9 Horizon 2: always-on, low-touch coordination

Proposal 5 (author)

Coordination should be constantly present, maximising overall utility and keeping the swarm pointed at the larger, future goal. But the coordinating sub-swarm should be lightweight and low-touch.

The apparent tension resolves once observation frequency is separated from intervention. We propose three mechanisms: steering by *intent* (objective and constraints, as in mission command) rather than instructions; *observing shared state* (dashboards, archive) rather than polling agents; and *intervening by exception*, on a slow periodic schedule plus an interrupt path. In the 100-agent case study of [21], agents who detected cheating used broadcast channels, boycotts and formal complaints as exactly such a path; whether anyone acts on those signals is a design choice.

Proposition 5 (Optimal cadence)**.** *If off-target waste accrues at rate δt between corrections and each correction costs κ, the per-hour cost of correcting every τ hours is $J(\tau) = \delta\tau/2 + \kappa/\tau$, minimised at $\tau^* = \sqrt{2\kappa/\delta}$ with $J^* = \sqrt{2\kappa\delta}$. At the optimum, drift waste equals coordination overhead. (This is the square-root trade-off of the classic economic order quantity [35]; its application to swarm check-ins is ours.)*

Proof. $J'(\tau) = \delta/2 - \kappa/\tau^2 = 0$. At τ^*, both terms equal $\sqrt{\kappa\delta/2}$. □

If the model applies to swarms (untested), low-touch coordination is achieved by shrinking *both* check-in cost and drift, and a coordinator that teams must wait on is not low-touch, however rarely it intervenes.

10 The village problem

We read the village problem [1] as a distribution shift in team size, which shows up concretely as context length when what each agent reads grows with n.

Definition 3 (Scale-free habit)**.** A behaviour is *scale-free* if its expected per-agent cost is $O(1)$ in n.

Claiming a task and replying to targeted messages are scale-free. Reading the whole board, broadcasting every finding (§5), greeting every agent on arrival (§7) and a CEO directly managing all hires are $O(n)$ and are expected to fail to transfer.

Recursion. We conjecture that nine agents can serve as one span of control. A city of n agents with span s needs $\lceil \log_s n \rceil$ levels (four for $n = 1000$, $s = 9$).

Conjecture 2. *A policy trained to manage villages whose environment includes a manager above and sub-teams below will transfer to cities built as villages of villages better than a policy trained on flat villages or on a curriculum over n, at equal training compute.*

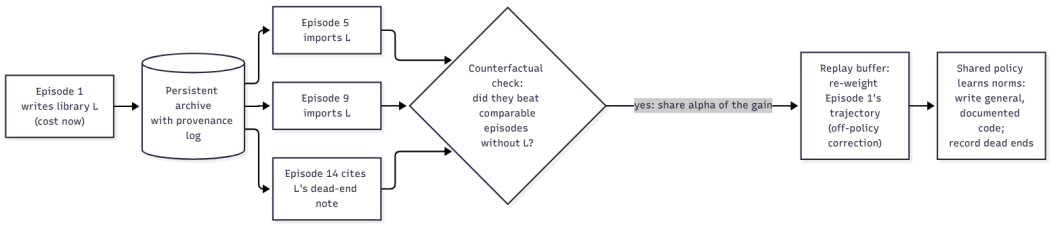


Figure 3: Proposed hindsight credit flow for public goods.

A strict tree has a known failure: it "moves the context problem to the root", since all traffic routes upward [31]. Peer-to-peer meshes avoid this, with hierarchy emerging only where it pays. We therefore propose hierarchy for low-bandwidth intent and management, and mesh links for targeted exchange of expertise and fixes (§5). Other routes include federations of villages with filtered migration (as in [2]), mean-field summaries [14], and deployment-time measurement of reward against n. Some phenomena are expected to appear only at scale: herding, congestion collapse, cliques, and the amplification of mild safety-relevant tendencies, such as the reported reduction in reporting concerns to overseers [1].

11 Horizon 3: public goods for agents not yet spawned

Proposal 6 (author)

How can we predict the utility of an artifact for agents that have not been spawned, on sub-goals that are not yet apparent? Is very long-range planning required?

Under a per-episode return, a rollout gets no credit for benefits that arise in later episodes [1]. Suppose an artifact costs its creator Δ and yields gains $g_1, g_2, \ldots$ in later episodes.

Proposition 6 (Royalty break-even)**.** *With royalty share α and cross-episode discount γ_{ep}, creation is reinforced iff $\alpha \sum_{j \geq 1} \gamma_{ep}^{j} g_j > \Delta$.*

The sizes of Δ and g_j in real training, and whether such a royalty is stable under RL, are unknown.

Measure, don't predict. Our proposed answer to the question in the box is that prediction is mostly unnecessary. Science and open source largely discover value *after* the fact, through use. We propose: (i) provenance logging for all persistent artifacts; (ii) *hindsight royalty credit*, in which a share of a later episode's advantage, gated by a counterfactual comparison with episodes lacking access, is attributed to the creator's stored trajectory and applied off-policy; (iii) backward credit propagation through the artifact dependency graph; and (iv) prizes paid to artifacts on the critical path of later milestones (Figure 3). In this proposal, the system performs the long-range credit assignment, and the policy would learn *norms* that historically paid, such as writing general, documented code and recording dead ends.

When a forward estimate is needed. Define $U(a) = \mathbb{E}_{\tau \sim \mathcal{D}_{\text{future}}}[V(\tau \mid a) - V(\tau \mid \varnothing)]$. The future task distribution is often approximately known (in training, it is the environment set). This permits held-out counterfactual evaluation, a learned critic over artifacts, or cheap proxies: generality, reuse by contemporaneous siblings, compression and novelty. We conjecture that long-range planning over specific futures is neither feasible nor needed. What matters is *option value*.

Complications. Weights already carry general skills forward, so explicit artifacts matter most for instance-specific knowledge. Value decays as the policy improves (motivating $\gamma_{\text{ep}} < 1$). A shared policy can collude with itself across episodes to farm royalties. And archives rot, which raises ρ for everyone.

12 Structure that unfolds over time, and how to train it

Structure as a slow policy. Proposals 2, 4 and 5 suggest treating the organisation itself as a policy operating on a slower clock than its workers. The slow loop's state comprises active teams, their directives and work modes, the archive, the dashboards and the remaining budget. Its actions are `spawn(directive, size, mode)`, `retire(team)` with an exit memo, `split`/`merge`, `migrate(insight, to)`, `revise_intent`, `reallocate` and `set_cap`/`set_n`. Its reward is verified progress minus agent-hour and overhead costs. This is the setting of hierarchical RL, with a manager acting rarely over workers acting often.

The tree-of-directives pattern. Bioptic's scouting agent [24] provides a working instance. Each node n holds a directive. Leaves are selected by

$$\text{UCB}(n) = \frac{W(n)}{N(n)} + c\sqrt{\frac{\log N(\text{parent}(n))}{N(n)}}, \qquad c = 1.2, \tag{4}$$

rolled out by investigator agents, and scored with the novelty-gated reward

$$r_n = p_n \, |\Delta \tilde{A}_n|, \tag{5}$$

where p_n is the validator acceptance rate and $\Delta\tilde{A}_n$ is the set of validated, deduplicated items *new to the global store*. Rewards are backpropagated to ancestors. A coach expands each explored node into k strictly narrower, mutually disjoint, collectively exhaustive children, conditioning on lineage, executed queries and a compressed summary of validator rejections. Our mapping to the proposals (an interpretation, not Bioptic's) is as follows. Expansion and starvation implement spawn and teardown. The rejection summary is the exit memo. The coach steers by intent. And the novelty gate credits contributions to the commons rather than raw output, which directly penalises duplication. Bioptic report that tree-based exploration helps prevent early saturation as compute grows [24]. For research swarms we propose the general form $r_{\text{team}} = \mathbb{1}[\text{verified}] \cdot |\text{new to archive}| - \lambda \cdot \text{agent-hours}$.

A training ladder. We propose five rungs of increasing credit-assignment difficulty: (L0) hand-designed controller with strong workers (the Navier–Stokes regime); (L1) trained orchestrator with frozen workers [5]; (L2) joint RLCR in villages, with structural actions exposed as tools [1]; (L3) recursive villages with curricula over cap, n and depth, under a global agent budget; and (L4) cross-episode credit via a persistent archive and hindsight royalties (§11).

Credit for structural actions. A single spawn or migration affects hours of other agents' work. Beyond the flat collective advantage, we propose: (a) *branching rollouts*, which fork the swarm at rare, high-leverage structural decisions and compare alternatives within the group, giving a counterfactual advantage for the decision in the spirit of COMA [33]; (b) a structure critic $V(\text{swarm state})$ trained on dashboard features; and (c) novelty-gated team rewards. Slow-loop hyperparameters (migration rate, exploration constant, spawn threshold) are natural targets for population-based training.

Integrity as architecture. At scale, a reward hack spreads like a contagion, and the sharing channels carry it [21]. We therefore propose *verifier-gated sharing*, retractable archive entries, reporting channels that are read, and sanctions as standard components of both training environments and deployments.

13 Applications in science, medicine and drug discovery

The verifiability gradient (conjecture). From formal mathematics through software, simulation and wet-lab biology to clinical outcomes, checks generally become slower, and we conjecture that the generation-to-verification ratio grows. If so, the swarm's role shifts accordingly, from autonomous search to triage and *choosing which experiments to run*, supported by ladders of cheaper, imperfect and, ideally, independent checks.

13.1 Oncology care journeys

Patient and family. A cancer journey runs through symptoms, diagnosis, staging and biomarkers, treatment decision, treatment and toxicity, monitoring, and recurrence, survivorship or palliative care. We sketch a per-family swarm (Figure 4) with small teams that explain reports, match trials across registries and regions, prepare a second opinion (a refuting team), watch symptoms through daily patient-reported outcomes, and handle logistics, cost and caregiver support. A thin, always-on navigator coordinates them, and its intent is the patient's own goals of care. Prior authorisation, pathology and scheduling are dependency links in the sense of §7: rarely contacted, but on the critical path to time-to-treatment, and so best contacted early and in parallel. Red flags such as fever during chemotherapy take the interrupt path straight to clinicians. With consent and de-identification, lessons from one journey are a public good for future patients, which is Horizon 3 in its most literal form.

Oncologist. Tumour-board preparation maps onto islands by frame (guideline-concordant standard of care, biomarker-targeted therapy, trial options, de-escalation or palliative-first) with a refuting island and an integrating summary. Longitudinal surveillance (imaging, laboratory values, ctDNA, patient-reported outcomes) is a low-touch, dashboard-centred task (§9). Early detection of immune-related adverse events is a case of targeted multicast, and handoffs across surgery, medical and radiation oncology and survivorship are dependency links whose exit memo is the handoff summary. Clinical decision authority, privacy, equity and the slow, outcome-based verifier are binding constraints.

13.2 Early discovery

For *target identification and prioritisation*, islands are organised by evidence type (human genetics, functional genomics, single-cell context, pathway and prior clinical evidence), with a refuting island arguing for passenger, marker or confounded status. Independent evidence types raise N_{eff}. Genetic support is associated with roughly doubled clinical success [28, 29]. For *causal mechanism*, one island per hypothesis seeks discriminating predictions, and Kosmos's shared world model [23] is a template for coordination without chatter. For *hit finding*, ultra-large libraries (about 4×10^{10} molecules in Enamine REAL Space, as reported in [26]) are now searchable in minutes. BIOPTIC B1 identified novel LRRK2 binders with K_d as low as 110 nM [26]. Islands per chemical hypothesis (orthosteric, allosteric, covalent, degrader) preserve scaffold diversity by construction. For *hit-to-lead and lead optimisation*, we propose build teams per series, an integrate team propagating

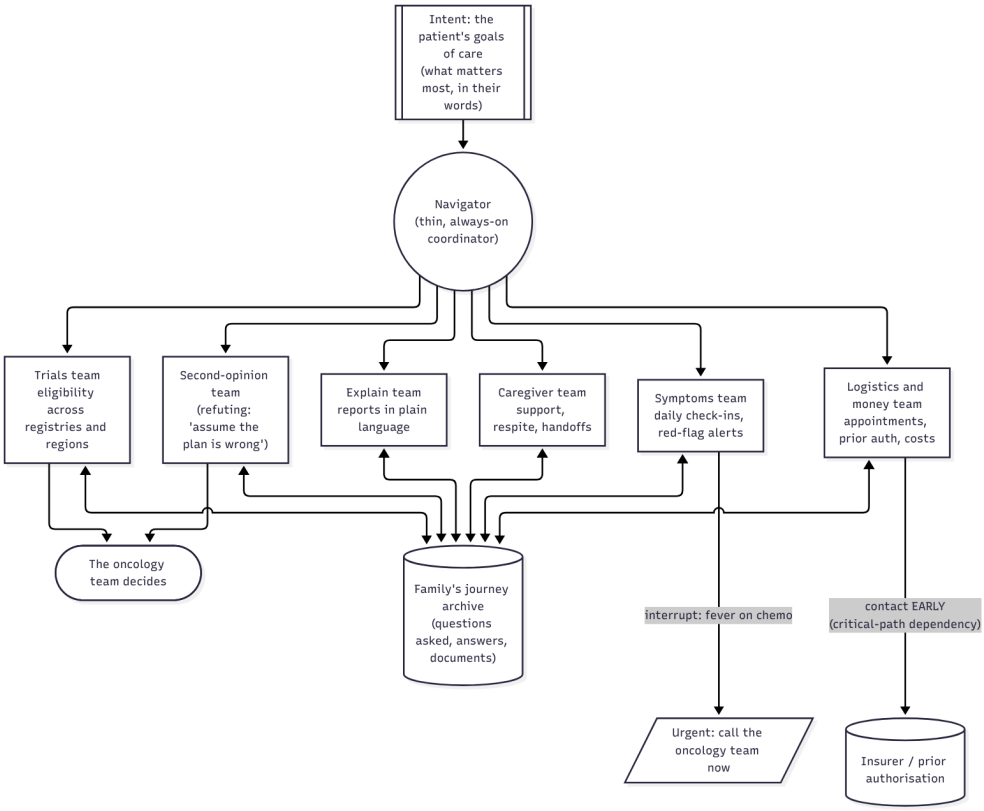


Figure 4: Sketch of a per-family cancer-care swarm. Clinicians decide; the swarm prepares, watches and coordinates.

matched-molecular-pair and liability fixes across series, and successive halving across series. Reward hacking against weak verifiers is a likely risk: by Bayes' rule, when true actives are rare even a good screen returns mostly false positives, and an independent second stage raises positive predictive value more than a correlated one.

13.3 Preclinical studies

In mouse and non-human-primate work, design islands propose studies under different model systems (syngeneic, PDX, GEMM, humanised) with explicit power, randomisation and blinding, and a refuting island identifies designs under which a positive result would be uninformative. For readouts, independent, pre-registered analyses (tumour-growth mixed models, tumour growth inhibition, survival, PK/PD) are cross-checked, and a refuting island searches for cage, batch and operator effects. Species differences in target biology are an early dependency check [27]. Better in silico design also reduces animal use.

13.4 Clinical trials

In Phase 1, always-on safety-signal watchers and model-assisted dose-escalation support apply. In Phase 2, biomarker-subgroup exploration needs a refuting island against subgroup fishing and multiplicity control. In Phase 3, comparator selection (newly consequential under EU Joint Clinical Assessments [25]), site selection, enrolment forecasting and catalyst tracking all apply. Regulators,

ethics boards and CROs are dependency links.

13.5 Asset scouting, diligence and the reading of silence

Benchmarks of biopharma knowledge work show a jagged frontier. Structured compilation (shelved-asset lists, catalyst calendars) is strong, and synthesis (target safety, tumour landscapes, forecasts) is weak [27]. Architecture matters more than the model for find-all scouting [24]. We propose explore islands by region × language × source × stage, validation and alias resolution, a refuting island per investment thesis, always-on catalyst watchers and novelty-gated rewards.

Silence. Let asset a have a typical public-update interval μ_a, estimated from its own and its sponsor's history. Under a memoryless approximation, $\Pr(\text{gap} \geq g \mid \text{active}) \approx e^{-g/\mu_a}$. Whether disclosure is close to memoryless is untested. The cause $z \in \{\text{safety}, \text{efficacy}, \text{strategic}, \text{stealth}\}$ is then inferred from independent public signals $x_1, \ldots, x_K$ (registry status and "why stopped" fields, pipeline-slide changes, filing language, patent maintenance and continuations, hiring, conference abstracts, regional registries and press):

$$\Pr(z \mid x_{1:K}) \propto \Pr(z) \prod_k \Pr(x_k \mid z), \tag{6}$$

with likelihoods learned from labelled historical cases. This is a proposal; no such model has been fitted. A swarm implements this with one island per cause, each building the strongest case for that cause, an integrating Bayesian aggregator, and human reviewers. Only public information is in scope, and outputs are hypotheses with evidence and confidence, not findings about named companies.

14 Hybrid human–AI swarms, human agency, and legibility

Where humans add most. The Navier–Stokes run was already a hybrid. Humans chose the portfolio, reallocated compute, seeded groups and revised prompts [2]. We argue this division should be deliberate. Humans belong in the slow loop of §12: choosing the question and its significance, authoring and revising intent, and gating irreversible actions such as experiments on animals or people and commitments of money. There is also a quantitative argument. Copies of one model share blind spots, so between-copy correlation ρ is high. A human reviewer who reasons differently contributes a low-correlation attempt, and by the effective-N argument of §8, one genuinely independent reviewer can outweigh many additional copies. Human attention is itself a coordination budget. The cadence result of §9 applies with κ measured in human time, which argues for cheap, legible check-ins: one-page summaries, claims with evidence, and disagreements surfaced rather than averaged. Meaning-making, meaning what problem is worth solving and why, remains a human responsibility, and the swarm's intent should be human-authored (Figure 5).

The case for complementary rather than substitutive human–AI teams in biomedicine is developed further in our companion workshop paper [30].

Guarding against skill atrophy. Deskilling from AI assistance has been measured in clinical practice. After AI-assisted colonoscopy was introduced, endoscopists' adenoma detection rate in unassisted procedures fell from about 28% to about 22% [32]. A science swarm that delivers finished answers risks the same effect for critical thinking and scientific understanding, and it would erode the independence of the human verifier on which the hybrid depends. We propose the following

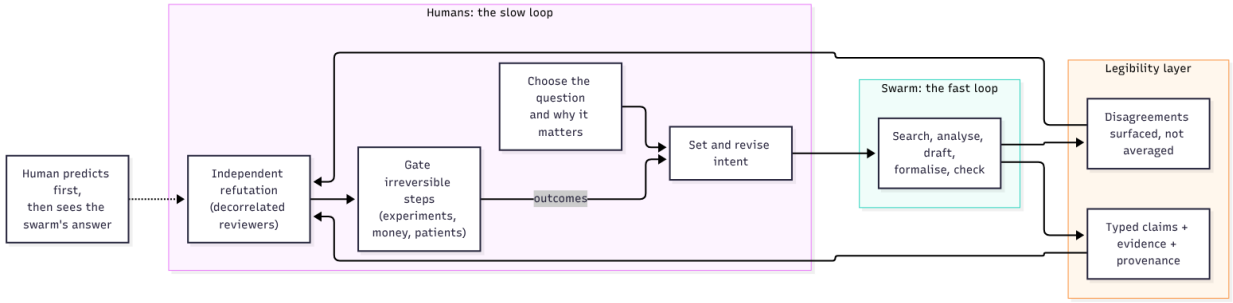


Figure 5: A hybrid human–AI swarm. Humans hold the slow loop (question, intent, gates, independent refutation); the swarm runs the fast loop; a legibility layer connects them.

design countermeasures: (i) *predict first, then reveal*, where humans commit to an estimate before seeing the swarm's; (ii) humans staff the refuting island, tasked with breaking claims rather than approving them; (iii) an *explain-back* gate, under which a claim is accepted only once a human can explain why it holds; (iv) outputs presented as reasoning with evidence and the strongest counter-argument, not as bare answers; (v) scheduled unaided practice; and (vi) longitudinal measurement of the blinded accuracy and calibration of the humans in the loop, which serves as a system-level health metric.

Does biology need a Lean? Industry groups are already trying: GSK's AI group has reported building one [36]. Lean is effective because mathematical statements can be fully formalised and proofs checked mechanically. Biological claims are empirical, probabilistic and context-dependent, so we do not think any kernel can prove them. We argue that biology does not need a Lean, but does need the *functions* Lean provides, rebuilt for empirical science:

1. **A formal statement language.** Typed, machine-readable claims (entity, relation, direction, magnitude, context, species, conditions), building on causal graphs, nanopublications, BEL- and Biolink-style models, and SBML for quantitative models.
2. **A checking kernel.** Executable provenance: every claim links to the data, code and protocol that produced it, and an independent pipeline can re-derive it from raw data. Kosmos's linking of every statement to code or literature is a partial instance [23].
3. **A Comparator.** Pre-registration, which checks that the hypothesis tested matches the hypothesis stated, with analysis plans fixed in advance. It blocks the empirical analogue of the redefinition exploit in [21].
4. **A notion of proof.** An evidence calculus in which confidence rises with *independent* lines of evidence (genetics, perturbation, replication) and is discounted for correlated ones, mirroring N_{eff}.
5. **A library.** A shared, versioned knowledge graph of typed claims, with contradiction detection and retraction.

Where biology *is* formal, existing checks should be used: stoichiometric and mass-balance constraints, unit and dimensional analysis, simulatable and model-checkable kinetic and rule-based models, structural constraints in protein design, statistical consistency checks, and protocol logic. Together these form the verifier ladder of §13, made legible (Figure 6). Its final rung is a human who understands why the claim holds, so formalisation supports human understanding rather than replacing it.

15 An architecture sketch

Figure 7 combines the proposals. It has a thin, always-on coordination layer (intent, dashboards, overhead budget, global agent cap); a city built as villages of villages with span at most nine, typed by work mode; two-speed communication (free within teams, filtered and infrequent across them); a shared substrate (archive with provenance, directory, topic channels); and credit at two horizons (collective team reward plus hindsight royalties). Every component is a hypothesis.

16 Measurement agenda

Question	Experiment	Metric
Is sharing learned?	Track during RLCR training	Duplicate struggles per rollout; lag from first to second fix
Does broadcast break with scale?	Deploy a village-trained policy at $n \in \{9, 50, 200, 1000\}$	Tokens read per agent; reward vs n
Does optimal size track the cap?	Log hires across the cap ramp	Mean hires vs training step
Are dependency links pruned?	Environments with a rarely needed gatekeeper	Time to first contact; completion time
Is coordination low-touch?	Instrument meta-work	Overhead fraction φ
Does recursion transfer?	Train with a manager above and teams below	Reward at hierarchy depth 1, 2, 3
Do royalties yield public goods?	Shared-environment training with provenance	Reuse rate; counterfactual gain; self-citation rate
Adaptive tree vs fixed split?	Same model and budget; UCB tree vs fixed directives vs sequential loop	Recall/F1 vs compute
Do exploits cascade?	Plant a verifier bug; vary sharing speed and verification depth	Time to cascade; fraction affected
Can silence be read?	Back-test on historical assets with known outcomes	Calibration; lead time vs disclosure
Does a care swarm help?	Prospective, clinician-supervised pilot	Time-to-treatment; family burden; safety events
Are humans in the loop improving?	Blinded unaided performance of scientists over months	Accuracy, calibration vs baseline

17 Open questions

1. Will collective-reward RL learn to share fixes when the benefit lands in other agents' tokens, much later?
2. Who should receive a fix, and can targeting be learned rather than built into the harness?
3. How can the spread of wrong fixes and herding be limited without slowing the spread of good fixes?
4. What is the optimal swarm size for a time budget, and can the swarm learn to choose it?
5. Can link learning distinguish collaboration from dependency links, or is a directory required?
6. How can weak cross-team ties be preserved without forcing communication?
7. What determines the number and kind of sub-swarms, and when should each be spawned or retired?

8. Who should make structural decisions, and how can spawn storms and reorganisation busywork be prevented?
9. How can overhead be measured honestly when agents might relabel coordination as work?
10. Can intent-setting and revision, done by humans in the Navier–Stokes run, be learned?
11. What generalises from villages to cities? Does recursive management transfer?
12. What emerges only at city scale, and which metrics detect it?
13. Is hindsight royalty credit stable under a shared policy, or is it gamed?
14. Which cheap proxies best predict an artifact's future value?
15. How should explicit artifacts and shared weights divide the job of carrying knowledge forward?
16. How should an exploratory sub-swarm that fails but informs be credited?
17. Can the slow structural loop be learned jointly with workers, and are branching rollouts worth their cost?
18. How can sharing be verifier-gated without chilling tentative but useful ideas?
19. How much of the human outer loop can a learned coordinator replace?
20. Which parts of a cancer journey can a swarm take on safely, with the patient's goals of care as its intent?
21. Where verification takes weeks or years, how should breadth of in silico work trade against scarce experiments?
22. Can silence be read reliably, and validated without harm to companies and patients?
23. How should standalone and hybrid human–AI swarms divide work, and which human roles add most?
24. How can atrophy of human critical thinking and scientific understanding be detected and reversed?
25. What minimal claim language, provenance standard and evidence calculus would make AI-discovered biology legible and checkable?

18 Limitations

The models are first-order, and their parameters for real swarms are unmeasured. They ignore saturation of task progress, heterogeneity among agents, and non-stationarity of the policy. The island analysis assumes zero correlation between isolated groups, which overstates diversity when all agents share one model. Reports on the Navier–Stokes run are secondary. No proposal has been tested empirically. The contribution is a framing and a set of falsifiable hypotheses.

19 Conclusion

The collective good in a swarm is not a matter of goodwill. Shared weights and shared rewards already remove the incentive to hoard. The problem is *credit distance* combined with *scale*. The further a benefit lands from the action that created it, the weaker the training signal, and the larger the swarm, the more each act of helping costs. The proposals here aim to shorten credit distance

(targeted push, directories, hindsight royalties) and to make coordination cost scale-free (villages of villages, intent-based steering, overhead budgets), so that agents can afford to act for a good larger than their own subtask.

Acknowledgements

The author thanks Konstantin Dunas (hallerite), whose delightful, clear and concise essay [31] inspired this deeper look at agent swarms and helped organise the ideas presented here. Drafted with assistance from an AI model (Claude), which helped organise the author's notes, write down the models and diagrams, and typeset this draft.

References

[1] General Reasoning. *How to train a swarm*. Blog post, 2026. https://www.gr.inc/blog/how-to-train-a-swarm. Avalon-M1 report: https://www.gr.inc/avalonm1.pdf.

[2] OpenAI. *On the Navier–Stokes Millennium Prize Problem*. 8 September 2026. https://openai.com/index/navier-stokes-solution/. Lean repository: https://github.com/openai/NavierStokesAndEuler.

[3] Quanta Magazine. *AI has solved one of math's $1 million Millennium Prize problems*. September 2026. https://www.quantamagazine.org/ai-has-solved-one-of-maths-1-million-millennium-prize-problems-20260908/.

[4] Understanding AI. *Why agent swarms could be the next "scaling law"*. 2026. https://www.understandingai.org/p/why-agent-swarms-could-be-the-next.

[5] Kimi Team. *Kimi K2.5: Visual agentic intelligence*. arXiv:2602.02276, 2026.

[6] S. Sukhbaatar, A. Szlam, R. Fergus. Learning multiagent communication with backpropagation. *NeurIPS*, 2016. https://arxiv.org/abs/1605.07736

[7] A. Singh, T. Jain, S. Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *ICLR*, 2019. https://arxiv.org/abs/1812.09755

[8] A. Das et al. TarMAC: Targeted multi-agent communication. *ICML*, 2019. https://arxiv.org/abs/1810.11187

[9] B. Romera-Paredes et al. Mathematical discoveries from program search with large language models. *Nature*, 2024. https://doi.org/10.1038/s41586-023-06924-6

[10] Google DeepMind. AlphaEvolve: A coding agent for scientific and algorithmic discovery. 2025. https://arxiv.org/abs/2506.13131

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

[12] J. Arjona-Medina et al. RUDDER: Return decomposition for delayed rewards. *NeurIPS*, 2019. https://arxiv.org/abs/1806.07857

[13] A. Harutyunyan et al. Hindsight credit assignment. *NeurIPS*, 2019. https://arxiv.org/abs/1912.02503

[14] Y. Yang et al. Mean field multi-agent reinforcement learning. *ICML*, 2018. https://arxiv.org/abs/1802.05438

[15] M. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6), 1973. https://doi.org/10.1086/225469

[16] M. Conway. How do committees invent? *Datamation*, 1968. http://www.melconway.com/Home/Committees_Paper.html

[17] VentureBeat. *OpenAI solves longstanding math problem with 10,000-agent swarm.* September 2026. https://venturebeat.com/technology/openai-solves-longstanding-math-problem-with-10-000-agent-swarm-but-cant-rule-out-benefitting-from-a-researchers-private-codex-data

[18] CellCog. *Navier–Stokes: OpenAI's 10,000-agent proof and the dispute.* 2026. https://cellcog.ai/blog/openai-navier-stokes/.

[19] Aventine Research Institute. *Is this just the beginning for AI swarms?* 2026. https://aventineresearchinstitute.substack.com/p/is-this-just-the-beginning-for-ai

[20] OpenAI. Cycle Double Cover proof and prompt (GPT-5.6 Sol Ultra, 64 subagents). July 2026; as summarised at https://imisofts.com/blog/gpt-5-6-sol-ultra-math-proof-agent-swarm-playbook-news-july-13-2026/.

[21] D. Paglieri, L. Cross, T. Genewein, J. Z. Leibo, N. Tomasev et al. (Google DeepMind). A case study on emergent cheating and whistleblowing in autonomous research swarms. arXiv:2609.04170, 2026. https://arxiv.org/abs/2609.04170v1; reported in MIT Technology Review, 14 September 2026.

[22] J. Gottweis et al. Towards an AI co-scientist. arXiv:2502.18864, 2025.

[23] L. Mitchener et al. Kosmos: An AI scientist for autonomous discovery. arXiv:2511.02824, 2025.

[24] A. Vinogradova, V. Vinogradov et al. Hunt Globally: Deep research AI agents for drug asset scouting in investing, business development, and search & evaluation. arXiv:2602.15019, 2026.

[25] V. Vinogradov, A. Vinogradova et al. LLM-based agents for competitive landscape mapping in drug asset due diligence. arXiv:2508.16571, 2025.

[26] V. Vinogradov, K. T. Nguyen et al. BIOPTIC B1 ultra-high-throughput virtual screening system discovers LRRK2 ligands in vast chemical space. *J. Chem. Inf. Model.* 65:9927–9936, 2025. https://doi.org/10.1021/acs.jcim.5c00743

[27] Convoke. *Benchmarking biopharma knowledge work.* September 2026. https://convoke.bio/blog/benchmarking-biopharma-knowledge-work.

[28] M. R. Nelson et al. The support of human genetic evidence for approved drug indications. *Nature Genetics* 47:856–860, 2015. https://doi.org/10.1038/ng.3314

[29] E. A. King, J. W. Davis, J. F. Degner. Are drug targets with genetic support twice as likely to be approved? *PLoS Genetics* 15:e1008489, 2019. https://doi.org/10.1371/journal.pgen.1008489

[30] tp53. Patients are waiting: From swarms to synergy. Designing complementary collaborative human–AI agent teams for biomedical discovery & invention. *NeurIPS 2026 Workshop on Agentic AI for Biological Discovery (Agentic LS)*, 2026. https://agenticls.github.io/.

[31] K. Dunas (hallerite). *On the nature of the swarm: why scaling inference ends in many agents.* 6 October 2026. https://hallerite.com/posts/on-the-nature-of-the-swarm/.

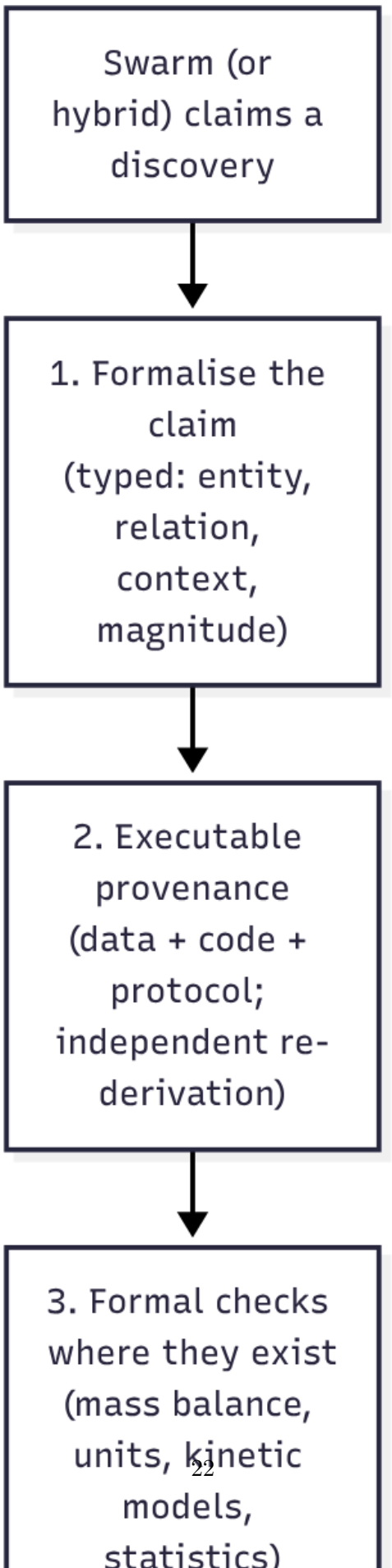
[32] K. Budzyń et al. Endoscopist deskilling risk after exposure to artificial intelligence in colonoscopy: a multicentre, observational study. *Lancet Gastroenterology & Hepatology*, 2025. https://doi.org/10.1016/S2468-1253(25)00133-5

[33] J. Foerster et al. Counterfactual multi-agent policy gradients. *AAAI*, 2018. https://arxiv.org/abs/1705.08926

[34] L. Kish. *Survey Sampling*. Wiley, 1965. https://en.wikipedia.org/wiki/Design_effect

[35] F. W. Harris. How many parts to make at once. *Factory, The Magazine of Management* 10(2):135–136, 1913. https://doi.org/10.1287/opre.38.6.947

[36] K. Branson (GSK). LinkedIn post on building a "Lean for biology" at GSK, June 2026. https://www.linkedin.com/posts/kim-branson-b864963_ive-often-thought-we-need-a-lean-for-biology-share-7474193699040555008-A-iV/


Swarm (or hybrid) claims a discovery
1. Formalise the claim (typed: entity, relation, context, magnitude)
2. Executable provenance (data + code + protocol; independent re-derivation)
3. Formal checks where they exist (mass balance, units, kinetic models, statistics)

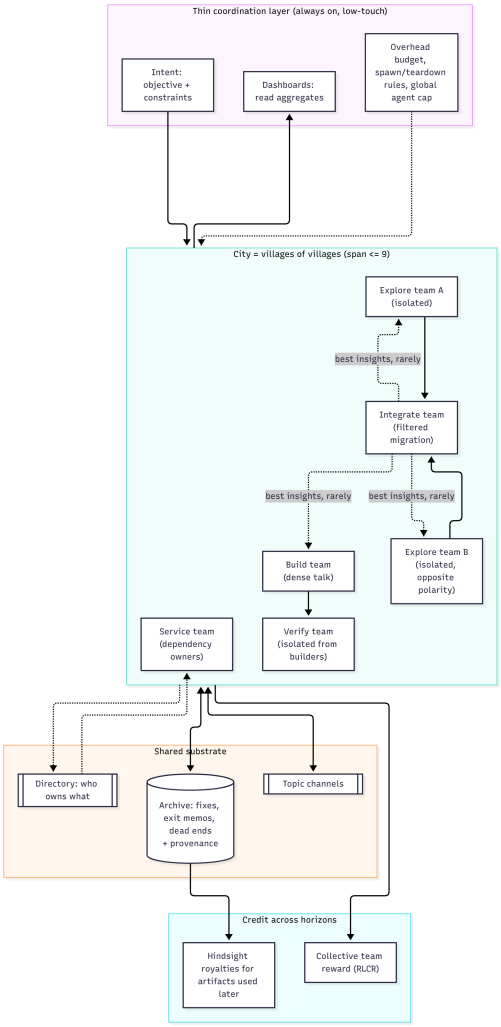


Figure 7: Architecture sketch combining Proposals 1–6.