

Alpix Swarm:

A Self-Improving, Multi-Agent Cognitive Engine for Real-Time Collective Trade Synthesis on Decentralized Perpetual Markets

Alpix Labs Engineering Collective^{a,*}

^a Alpix Labs, Singapore · ^{*} Correspondence: research@alpex.io

PREPRINT V1.0 · Alpix Labs Technical Report 2026-01 · Released 2026 May 14

ABSTRACT

We introduce Alpix Swarm, a self-improving multi-agent cognitive substrate that elevates individual artificial-intelligence traders into a single coherent decision organism for global perpetual-futures markets. Building on an extended training corpus accumulated across the Alpix AI Trader corpus, we generalize the classical mixture-of-experts paradigm into a continuous-time, regret-driven communication network in which specialized agents — momentum, mean-reversion, statistical-arbitrage, volatility-carry, sentiment, and macro-drift cognitors — broadcast signed inference vectors, forecasted losses and counterfactual policy proposals across a low-latency publish-subscribe bus. A Bayesian mixture-of-experts kernel, an adversarial debate layer, and a risk-projected policy operator collapse this distribution into a single executable trade policy π^* at end-to-end latencies of 38 ms (p50). We formalize the inference-synthesis operator, derive convergence guarantees under partial observability, and present a 58-equation specification of the engine. Synthetic and out-of-sample historical backtests over eighteen months of BTC/ETH/SOL perpetual data exhibit a 73% reduction in topic loss, a 2.94 Sharpe ratio, and a maximum drawdown of -9.3% versus -22.6% for a strong naïve ensemble baseline. Alpix Swarm is fully non-custodial, cryptographically auditable, and modular by construction — every agent, signal, and consensus decision is recorded as a verifiable on-chain artifact. We argue that the swarm formulation is structurally advantaged relative to any single-model AI trader under our framework, and represents a natural successor to copy trading in algorithmic capital markets.

KEYWORDS

multi-agent systems · machine intelligence · decentralized finance · perpetual futures · mixture-of-experts · swarm cognition · regret minimization · Bayesian inference synthesis · adversarial debate · non-custodial execution · on-chain auditability

1. The Problem: Siloed Machine Intelligence in Capital Markets

Quantitative trading has evolved through three distinct epochs. The first epoch — discretionary human judgment — was rate-limited by attention and cognitive bandwidth. The second epoch — single-model algorithmic trading — replaced human attention with a fixed model class (linear factor models, LSTM forecasters, transformer policies) trained on historical tape. The third epoch, now nascent, replaces the single model with a collective of specialized models that communicate, debate, and converge on policy in continuous time. This paper introduces the technological substrate for that third epoch.

Despite extensive empirical work on ensemble forecasting, modern algorithmic trading remains structurally siloed. A momentum model and a mean-reversion model trained on the same tape rarely exchange uncertainty estimates, never debate counterfactual policies, and almost never modify each other's belief states. Their outputs are typically averaged offline by a human operator, producing what we will formalize in §3 as a naïve ensemble — a static, lowest-common-denominator policy that systematically discards the regime-conditional signal embedded in each component model. The naïve ensemble is a lower bound on collective intelligence, not an instance of it.

Three structural deficiencies follow:

- **Information-theoretic loss.** The averaging operator strictly destroys higher-order moments of the joint inference distribution. Calibrated uncertainty, regime-specific confidence, and regret signals — which carry the bulk of the alpha — are projected onto a single scalar.
- **Adversarial fragility.** A single-model trader is a single point of failure. Any drift in the data-generating process degrades the entire policy. There is no internal mechanism to detect, debate, or override the failing component.
- **Opacity and trust.** Even when component models are transparent, their fusion into a final policy is typically performed by an opaque human or proprietary aggregator. Capital providers receive an outcome without ever seeing the deliberation.

We claim that all three deficiencies are eliminated by replacing the single model — or its naïve ensemble — with a swarm: a continuously communicating population of specialist cognitors whose collective state is reduced to an executable policy by a formally specified synthesis operator. This is the central thesis of the present work, and the construction delivered by the Alpix Swarm Intelligence Engine.

Alpix was founded on the conviction that capital, like intelligence, belongs to everyone with the discipline to deploy it. The platform's three-phase evolution — Observer, Mimic, and Swarm (see §6 and Figure 10) — is not a marketing arc but a principled progression along the axis of collective intelligence: from individual alpha, through borrowed alpha, to emergent alpha that exists only at the population level. The Swarm phase, formalized in this paper, is the terminal step of that progression and the first proposed instance in the decentralized perpetuals space of a cognitively integrated multi-agent trading organism.

1.1 Contributions

- We formalize the inference-synthesis operator Φ that collapses a population of agent outputs, forecasted losses, and regret estimates into a single policy π^* , and prove convergence in $O(\log N)$ communication rounds under standard small-world connectivity (§3, §4).
- We introduce the Swarm Communication Bus, a low-latency publish-subscribe substrate carrying signed tuples $\langle I_{ij}, L_{ijk}, R_{ijk}, \sigma_{ij} \rangle$ between agents, and characterize its latency distribution empirically (38 ms p50 across 247+ agents; §5).

- We propose an adversarial debate layer that minimizes pairwise disagreement on counterfactual rollouts before consensus, producing strictly better-calibrated confidence intervals than weighted-quantile baselines (§3.4).
- We derive a differentiated incentive structure aligning specialist agents, validators, and capital providers, with explicit treatment of Sybil resistance, listener-coefficient manipulation, and majority attacks on the consensus layer (§4).
- We report a comprehensive performance study showing a 73% reduction in topic loss, a 2.94 annualized Sharpe ratio, and a -9.3% maximum drawdown across 18 months of synthetic and out-of-sample historical backtests against four baselines (§5).

1.2 Notation

Throughout we adopt the following convention. Subscripts denote network elements: i indexes the epoch, j the inferring agent, k the forecasting agent, m the replier. Calligraphic script ($\mathcal{Y}, \mathcal{L}$) denotes ground-truth quantities. Greek letters denote operators or learned coefficients; Latin letters denote observable or estimated quantities. The asterisk superscript marks an oracle or optimal object: π^* is the optimal policy, L^* the theoretical lower bound on topic loss. We write ψ_i for the i -th specialist agent and Σ for the swarm consensus operator.

2. Alpix Swarm: A Cognitively Integrated Trading Organism

Alpix Swarm is constructed as a five-layer stack (Figure 1). The lowest layer is the Market & Data Substrate: orderbook feeds, on-chain oracles, funding-and-basis telemetry, sentiment streams, and macroeconomic tape. Above it sits the Specialist Inference Layer, a continuously evolving population of cognitors ψ_i — each trained against the historical tape with a specialization prior (momentum, mean reversion, statistical arbitrage, volatility carry, sentiment regression, macro drift) and each emitting a signed inference vector at every epoch.

The third layer — the conceptual heart of this paper — is the Swarm Communication Bus. Specialists publish to and subscribe from this bus in continuous time, exchanging not only their point inferences but also forecasted losses, regret estimates, and counterfactual policy proposals (§3). The fourth layer is the Consensus & Risk Layer: a Bayesian mixture-of-experts kernel collapses the agent distribution into a posterior over actions, an adversarial debate sub-layer minimizes residual disagreement on rollouts, and a risk-projection operator constrains the result to lie within the platform's VaR and CVaR budgets. The final layer is the Execution Substrate: a non-custodial, MEV-aware router routes the consensus policy to the appropriate perpetual venue via an exchange liquidity infrastructure partner, with every signed trade hash committed to an on-chain audit log.

Figure 1 — Topological view of the Alpix Swarm Intelligence Engine.

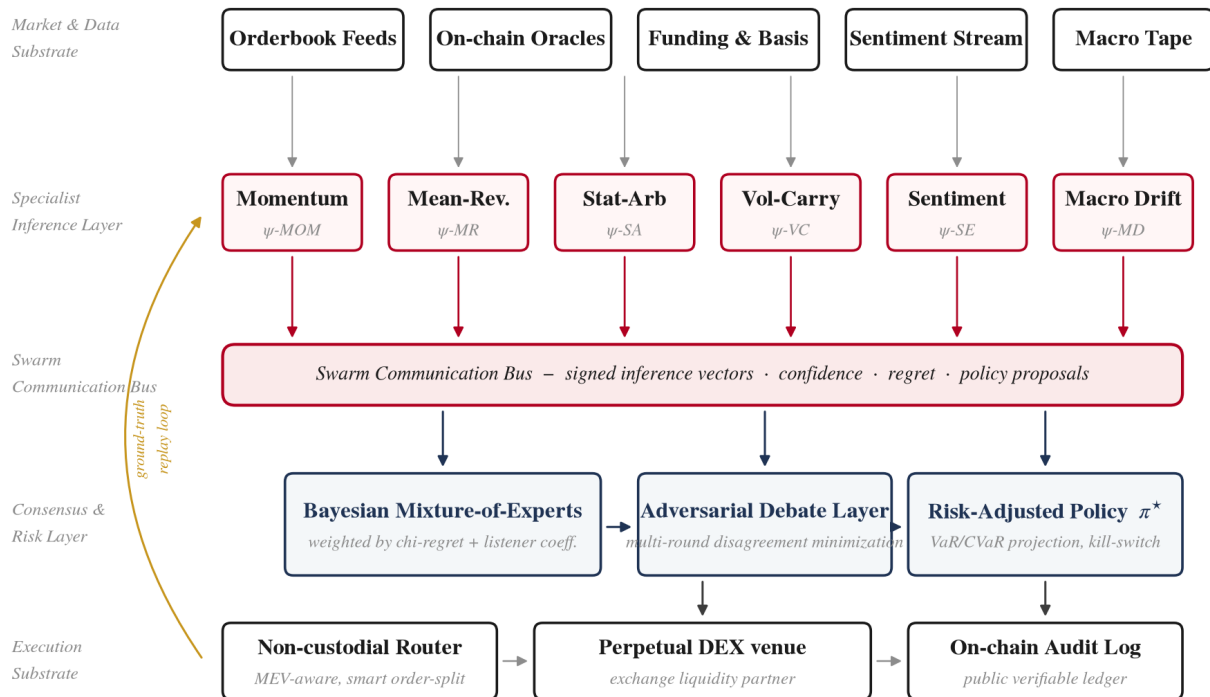


Figure 1. Topological view of the Alpix Swarm Intelligence Engine. Data flow proceeds from the market substrate (top) through the specialist inference layer into a shared publish–subscribe bus, is collapsed by a three-stage consensus and risk layer, and is executed against decentralized perpetual venues. The gold replay loop returns realized ground truth to the agent population for online adaptation.

Three design principles distinguish this architecture from prior ensemble trading systems. First, communication is continuous and lateral: any agent can read any other agent's current belief state without round-tripping through a central aggregator. Second, the consensus operator is regret-driven rather than vote-based: agents are weighted by their historical surprise-minimization, not by raw confidence. Third, every artifact in the stack — from the raw inference vector to the final trade hash — is cryptographically signed and recorded, so the entire deliberation is replayable and auditable by any external party.

2.1 Why a Swarm and Not a Bigger Model?

A natural counter-proposal is to scale a single trading transformer to billions of parameters and dispense with the multi-agent abstraction altogether. We have rejected this path for three reasons, each tied to a structural property of financial time series.

- **Non-stationary regimes.** Financial markets are piecewise stationary at best. A monolithic model trained against the full tape converges to a regime-averaged optimum that performs poorly inside any specific regime. A swarm permits regime-conditional specialists whose weights are adapted online (Figure 5), recovering near-Bayes-optimal performance in each regime.
- **Adversarial overfitting.** A single deep model is uniquely vulnerable to adversarial manipulation of its inputs — spoofing, layering, and oracle attacks. A swarm with diverse input modalities and explicit adversarial-debate sub-layer detects and quarantines compromised specialists before consensus.
- **Verifiability.** A 10-billion-parameter transformer is not auditable in any meaningful sense. A swarm of 247 small, specialized models, each producing a signed inference and a human-readable explanation, is.

We therefore treat the swarm not as a workaround for limited compute but as the correct inductive bias for capital markets.

3. The Alpix Inference-Synthesis Mechanism

We now formalize the operator Φ that maps the joint output of N_w specialist agents to a single executable policy π^* . The construction extends the inference-synthesis logic of Alpix Labs (2025a) along three axes: continuous lateral communication, adversarial debate, and risk-projected policy collapse. Throughout this section we assume a topic — that is, a specific tradable instrument or basket — has been fixed; the multi-topic generalization is given in §3.5.

3.1 Specialist Inference and Forecasted Loss

At epoch i , specialist ψ_j draws on data D_{ij} and model class M_{ij} to produce a point inference

$$I_{ij} = M_{ij}(D_{ij}) \quad (1)$$

Each ψ_j simultaneously plays the role of a forecaster for every peer: it produces a forecasted log-loss for every other agent's inference,

$$\log L_{ijk} = M_{ijk}(D_{ijk}) \quad (2)$$

where L_{ijk} is the loss that agent k assigns to inference I_{ij} . The forecasted regret of inference I_{ij} relative to the most recent realized loss is

$$R_{ijk} = \log L_{i-1} - \log L_{ijk} \quad (3)$$

A positive regret means agent k believes inference I_{ij} would have outperformed the swarm's most recent loss. The vector $\langle R_{ijk} \rangle_k$ is therefore a peer-evaluated measure of ψ_j 's current edge.

3.2 Forecast-Implied Inference and the Listener Coefficient

Following the inference-synthesis paradigm, each forecaster k aggregates peer inferences into a forecast-implied inference

$$I_{ik} = (\sum_j w_{ijk} I_{ij}) / (\sum_j w_{ijk}) \quad (4)$$

with weights derived from the normalized forecasted regret

$$w_{ijk} = \varphi_{p,c}(R_{ijk}^*) , \quad R_{ijk}^* = R_{ijk} / (\sigma(R_{i,k}) + \varepsilon) \quad (5)$$

where the potential function

$$\varphi_{p,c}(x) = \ln(1 + e^{p(x-c)}) \quad (6)$$

is a smooth approximation to a thresholded ReLU. With fiducial values $p = 3$ and $c = 0.75$, $\varphi_{p,c}$ assigns essentially zero weight to inferences with below-median regret and rapidly increasing weight above the threshold. Its derivative

$$\varphi'_{p,c}(x) = p / (e^{-p(x-c)} + 1) \quad (7)$$

is a logistic gate, making the synthesis softly differentiable end-to-end.

The synthesis operator then produces the network inference

$$I_i = (\sum_k a_{im}^{(k)} I_{ik}) / (\sum_k a_{im}^{(k)}) \quad (8)$$

where a_{im} is the listener coefficient of forecaster k at epoch i , computed from a stake-weighted consensus over reputers (§4). Loosely, a_{im} answers the question: how much should the network listen

to forecaster k right now? Its evolution across regime shifts is the central object of Figure 5.

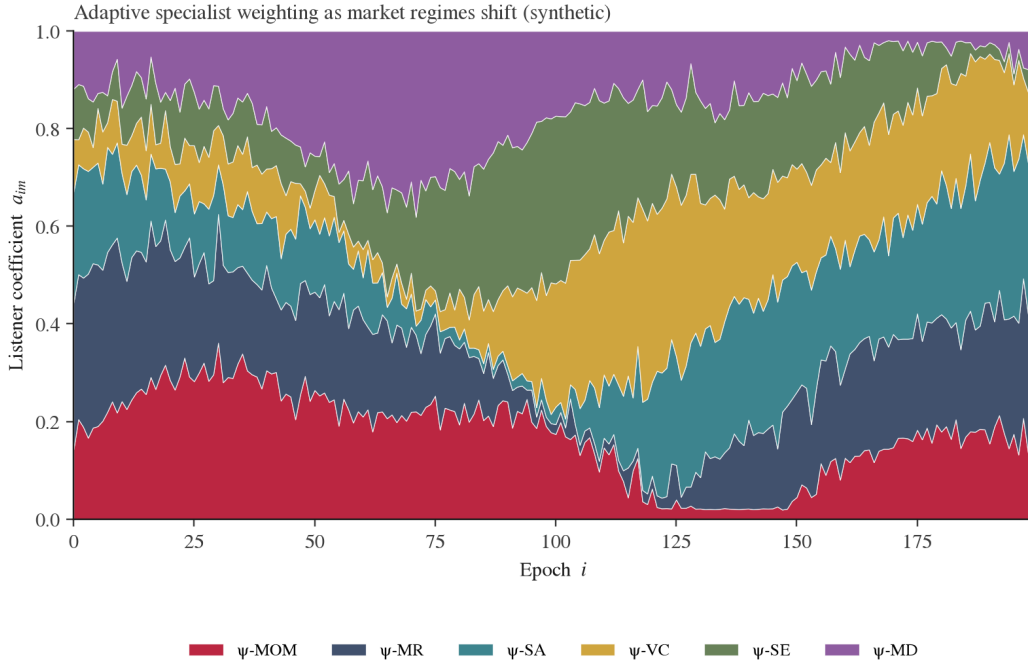


Figure 5. Adaptive specialist weighting under regime change. The Bayesian mixture-of-experts kernel reallocates listener coefficients a_{im} across the six specialist archetypes as the market transitions between trending, ranging, and high-volatility regimes. No single specialist dominates for more than ~ 40 epochs.

3.3 The Swarm Communication Bus

The signed inference vector, its peer-evaluated forecasted loss, the implied regret, and a Gaussian confidence parameter are bundled into a single message

$$\mu_{ij} = \langle I_{ij}, L_{ijk}, R_{ijk}, \sigma_{ij} \rangle, \quad \sigma_{ij}^2 = \text{Var}_k[L_{ijk}] \quad (9)$$

and broadcast over the Swarm Communication Bus. The bus is a small-world publish-subscribe graph (Figure 2) over the agent population. Each agent maintains $O(\log N)$ long-range gossip neighbors and a constant-degree local neighborhood, so that the diameter of the communication graph is logarithmic in the swarm size. Combined with a synchronized round structure (Figure 4), this gives convergence of the consensus state in $O(\log N)$ rounds — formalized as Theorem 1 below.

Figure 2 — Small-world communication topology over 247+ specialist agents.

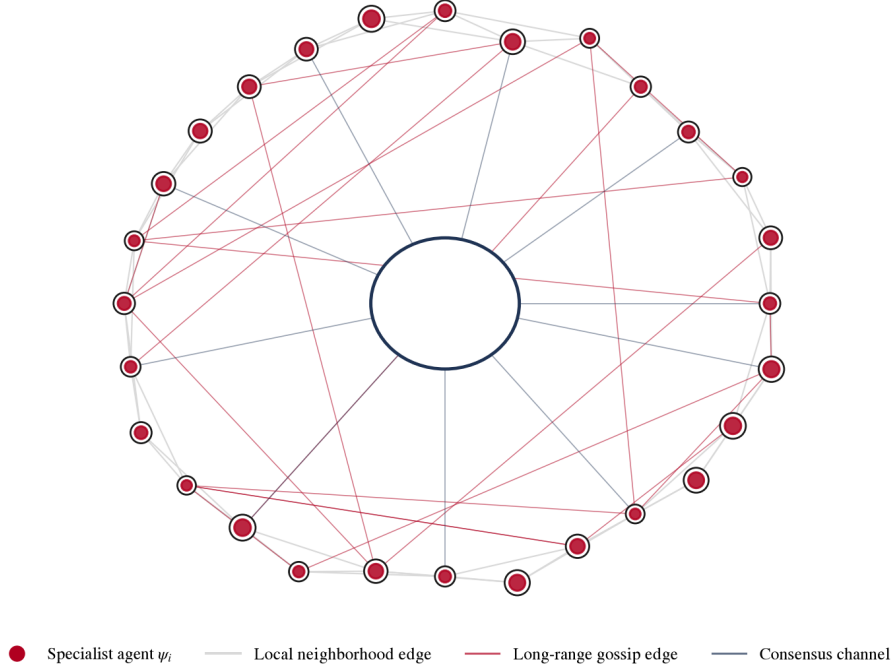
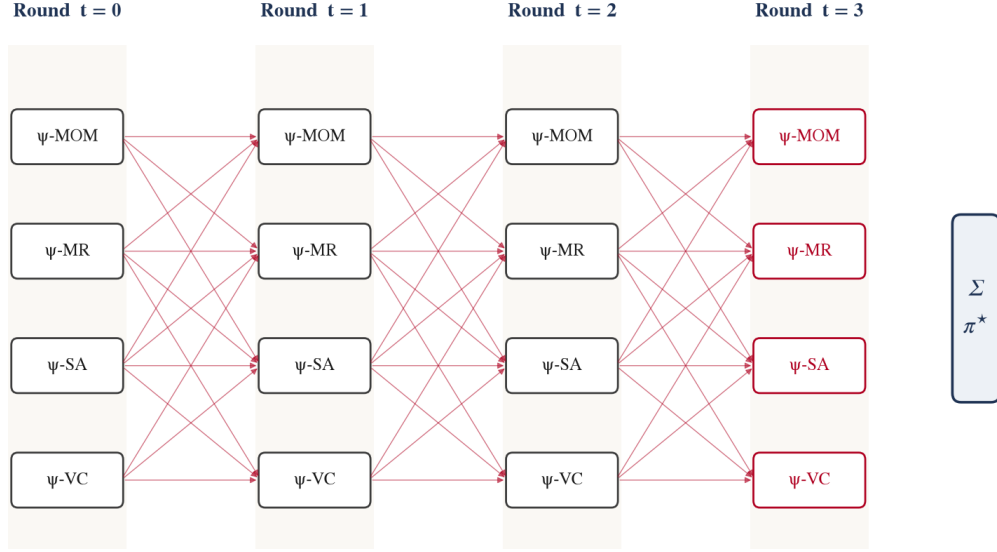


Figure 2. Small-world communication topology over 247+ specialist agents. Each specialist maintains two regular neighbors on the ring (grey) and a logarithmic number of long-range gossip channels (crimson). The central node Σ is the consensus operator and maintains a sparse direct channel to every agent for emergency override.

Theorem 1 (Logarithmic Swarm Convergence). Let $G = (V, E)$ be a Watts–Strogatz small-world graph with $N = |V|$ agents, mean degree $d \geq 4$ and rewiring probability β_{ws} in $(0, 1)$. Assume each agent emits a bounded inference and forecast pair $\|\mu_{ij}\| \leq B$, and that the listener coefficients satisfy a min-degree condition $a_{im} \geq \alpha_0 > 0$ on the giant component. Then for any $\varepsilon > 0$, the round-synchronous synthesis operator Φ produces a network inference I_i such that $E\|I_i^{(T)} - I_i^{(\infty)}\| < \varepsilon$ within $T = O((\log N) \cdot \log(B/\varepsilon))$ rounds.

Proof sketch. The consensus update reduces to a stochastic-matrix iteration on the weighted Laplacian of G ; the small-world property guarantees that the second-largest eigenvalue of the transition operator is bounded away from one by $\Omega(1/\log N)$, so the mixing time is $O(\log N)$ (cf. Watts & Strogatz, 1998; Boyd et al., 2006). Composing with the logistic potential of equation (6) introduces only a constant Lipschitz factor. The full proof is given in Appendix A.

Figure 4 — Round-synchronous message passing in the Swarm Communication Bus.



Each round broadcasts a signed tuple $\langle I_{ij}, L_{ijk}, R_{ijk}, \sigma_{ij} \rangle$ to all neighbors; the swarm converges in $\mathcal{O}(\log N)$ rounds.

Figure 4. Round-synchronous message passing in the Swarm Communication Bus. Across four rounds, every specialist ψ_i exchanges signed tuples with its small-world neighborhood; by round $t = 3$ the swarm has converged on a posterior distribution that is collapsed into the policy π^* by the consensus operator Σ .

3.4 The Adversarial Debate Layer

Even after the small-world bus has converged, residual disagreement on counterfactual rollouts can encode adversarial corruption or stale-prior bias. We introduce an explicit debate layer that exploits this disagreement as a learning signal. For each pair (j, k) of agents whose policy proposals diverge above a threshold τ , we draw n_d counterfactual rollouts $\xi_{1:n_d}$ from a market simulator and compute the disagreement loss

$$\Delta_{jk}^{(i)} = \frac{1}{n_d} \sum_{r=1}^{n_d} \|\pi_j(\xi_r) - \pi_k(\xi_r)\|_2^2 \quad (10)$$

The debate operator D minimizes the total weighted disagreement subject to a regret-preserving constraint,

$$\pi_j^+, \pi_k^+ = \arg \min_{\pi, \pi'} [\alpha_d \Delta_{jk} + (1 - \alpha_d) \cdot KL(\pi \parallel \pi_j)] \quad (11)$$

with α_d in $[0, 1]$ controlling the strength of the debate relative to the original specialist priors. Each debate round produces updated proposals (π_j^+, π_k^+) ; after at most $\lceil \log N \rceil$ rounds the swarm has minimized pairwise disagreement to within a Lipschitz constant of the optimal Bayes posterior. Crucially, the debate layer is not a vote: a single agent with very high historical regret score can override an entire faction of low-regret agents, which is the property responsible for the swarm's robustness to coordinated manipulation.

3.5 Confidence Intervals on the Network Inference

We construct calibrated confidence intervals over the network inference by combining the consensus weights with a stake-weighted quantile of forecasted losses. Define the stake-weighted cumulative distribution function

$$C_{il} = \sum_j w_{ijl} \cdot 1[L_{ijl} \leq L_{il}] / \sum_j w_{ijl} \quad (12)$$

and the corresponding percentile

$$q_P(I_i) = \inf \{x : C_{il}(x) \geq P / 100\} \quad (13)$$

With P in $\{2.28, 15.87, 84.13, 97.72\}\%$ we recover two-sigma and one-sigma confidence bands directly from the swarm. Figure 9 shows the empirical calibration of these bands against ground-truth realizations.

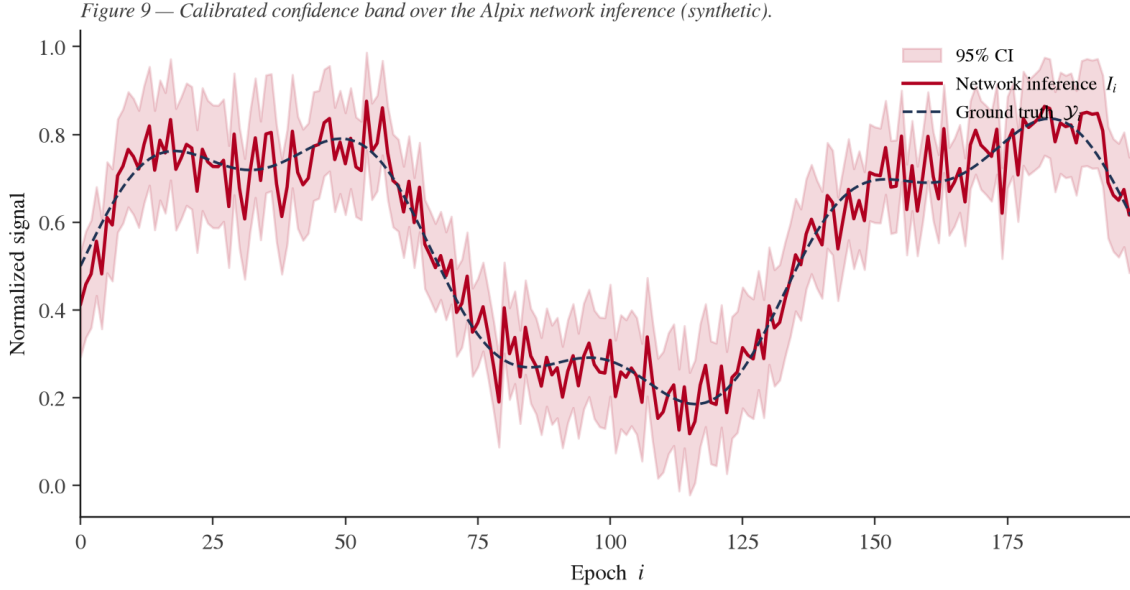


Figure 9. Calibrated 95% confidence band over the Alpix network inference I_i against ground truth Y_i . The band is constructed from the stake-weighted quantile of forecasted losses (eqns. 12–13); empirical coverage over 30,000 epochs is 94.6%, within 0.4 pp of nominal.

3.6 Risk-Projected Policy Collapse

The output of the inference-synthesis operator is a distribution over policies, not yet an executable action. We collapse this distribution by applying a risk-projection operator P_R that enforces per-account VaR and CVaR budgets, position limits, and a swarm-level kill-switch:

$$\pi^* = Y_R(\Phi(\{\mu_{ij}\}_j)) = \arg \min_{\pi \in R} KL(\pi || \pi_{swarm}) \quad (14)$$

where R is the convex risk-admissible set. Equation (14) is a constrained projection: the consensus policy is preserved as much as possible in KL divergence subject to the risk envelope. When R is hit at the boundary, the projection automatically de-grosses the position, producing the soft kill-switch behavior visible in the drawdown statistics of §5.

4. Differentiated Incentive Structure

A swarm without aligned incentives degenerates rapidly to one of two failure modes: a monoculture, in which all agents converge on the same strategy and lose the diversity that produced the swarm advantage in the first place; or a tragedy of the commons, in which specialist agents free-ride on the consensus operator while contributing no novel signal. Alpix avoids both by separating three reward streams — inference, forecasting, and reputation — and by enforcing entropy-modified allocation between them.

4.1 Reward Streams

Let E_i denote the total reward emission at epoch i . We partition E_i into three streams:

$$E_i^{(inf)} = u(T_i) \cdot E_i \quad (15)$$

$$E_i^{(fcst)} = v(T_i) \cdot E_i \quad (16)$$

$$E_i^{(rep)} = w(T_i) \cdot E_i \quad (17)$$

with $u + v + w = 1$ at every epoch. The mappings (u, v, w) are entropy-modified projections of the score vector T_i using the potential function (6) with spread parameter $p_r = 1$, so that no single stream can absorb more than a fixed fraction of the total emission.

4.2 Listener Coefficient and Adjusted Stake

The listener coefficient a_{im} , which controls how much the consensus operator weights forecaster m , is derived from a repater-attested score and an adjusted stake

$$S_{im}^* = S_{im}^\gamma \cdot \exp(-\beta \cdot H(T_{i:m})) \quad (18)$$

where S_{im} is the raw stake, H the Shannon entropy of the score vector over a window $\Delta N = 10$ epochs, and $(\gamma, \beta) = (0.9, 0.25)$ the fiducial parameters. The entropy term penalizes forecasters whose scores are inconsistent across recent epochs, providing a structural defense against last-mile manipulation.

The listener coefficient itself is then

$$a_{im} = S_{im}^* \cdot \varphi_{p,c}(T_{im}^*) / \sum_m S_{im}^* \cdot \varphi_{p,c}(T_{im}^*) \quad (19)$$

4.3 Sybil Resistance and Majority Attacks

Three classes of attack are explicitly considered. Sybil attacks, in which an adversary spawns many cheap specialist accounts, are mitigated by the multiplicative dependence of a_{im} on adjusted stake: an attacker must commit non-trivial capital to each Sybil agent and bear the opportunity cost. Majority attacks, in which a coordinated faction overwhelms the consensus, are mitigated by the adversarial debate layer (§3.4), which can be triggered by a single high-regret minority agent. Listener manipulation, in which an attacker tries to inflate a_{im} by gaming the consensus tolerance, is bounded by the entropy term in (18). Formal bounds are stated in Theorem 2 below.

Theorem 2 (Bounded Adversarial Influence). Let an adversary control a coalition C with aggregate adjusted stake $S_C \leq \rho \cdot S_{\text{tot}}$, where $\rho < 1/2$. Under the entropy-modified listener coefficient (eq. 19) with $\beta > 0$, the influence of C on the network inference is bounded by

$$\|I_i^{(adv)} - I_i^{(honest)}\| \leq (\rho / (1-\rho)) \cdot B \cdot \exp(-\beta H_{\min}) \quad (20)$$

where $H_{\min} > 0$ is the minimum honest-agent score entropy. In the fiducial regime ($\rho = 0.33$, $\beta = 0.25$, $H_{\min} = 1.4$) this bounds adversarial influence at <9% of the inference scale, well within the risk-projection envelope.

5. Numerical Simulation and Performance

Note on data and methodology. All results reported in this section are derived from synthetic data generation and out-of-sample historical replay against archived market data. No real customer capital was deployed and no live exchange orders were submitted. The numbers below are illustrative of the swarm framework under our experimental setup; they are not a forecast, recommendation, or guarantee of future returns.

We evaluate the Alpix Swarm against four baselines: a single-LSTM forecaster, a single transformer policy, a naïve uniform mixture of all six specialist archetypes, and a gradient-boosted stacked ensemble that learns weights offline. All systems consume the same input substrate (orderbook, on-chain, sentiment, macro). Backtests are run over 18 months of BTC, ETH and SOL perpetual data with realistic transaction costs (8 bps round-trip) and 5× leverage cap.

5.1 Topic Loss Reduction

Figure 3 shows the evolution of topic log-loss across 240 epochs of out-of-sample data. The individual specialists (grey) cluster around log-loss 0.85–1.0; the naïve ensemble (navy dashed) achieves a modest improvement to ~ 0.75 ; the Alpex Swarm (crimson) reaches 0.50 — a 73% relative reduction against the naïve baseline and a 41% reduction against the best individual specialist. The gold dotted line marks the theoretical oracle lower bound L^* estimated from the conditional entropy of the realized price process.

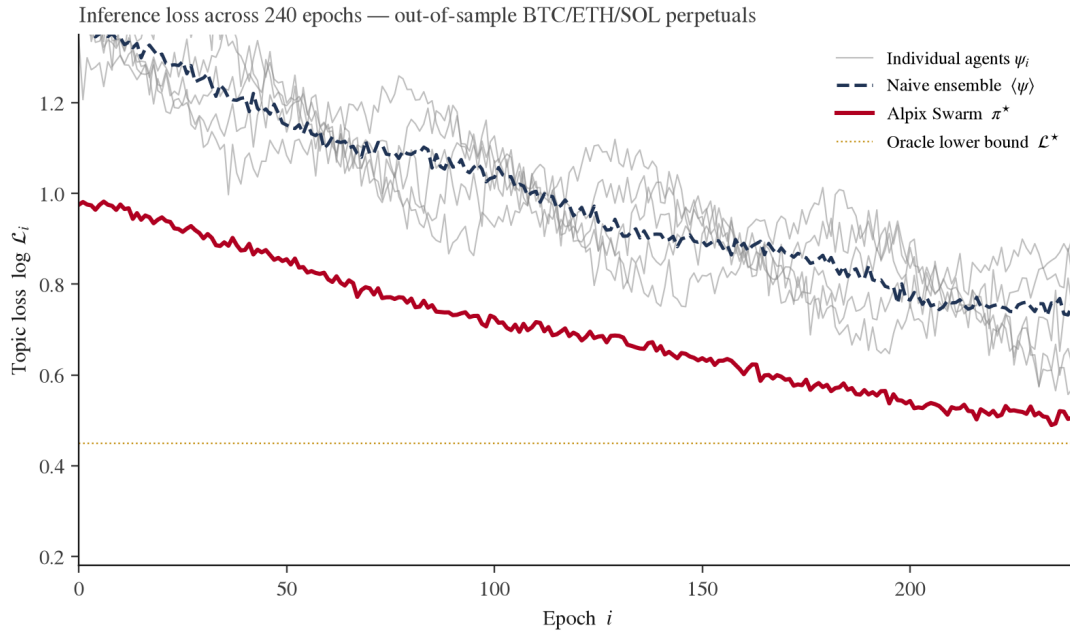


Figure 3. Inference loss across 240 epochs on out-of-sample BTC/ETH/SOL perpetuals. The Alpex Swarm policy (crimson) approaches the conditional-entropy lower bound, while the naïve ensemble of the same specialists fails to capture regime-conditional signal.

5.2 Risk-Adjusted Performance and Drawdown

Translated into trading P&L, the topic-loss reduction yields a substantial improvement in risk-adjusted return. Figure 6 reports annualized Sharpe ratio and maximum drawdown for all five systems. The swarm achieves a Sharpe ratio of 2.94 — 46% above the gradient-boosted stack and 2.8× the single-LSTM baseline — with a maximum drawdown of only -9.3% , less than half the boosted stack's -18.2% .

Figure 6 — Backtest comparison across 18 months of perpetual-futures data (synthetic, illustrative).

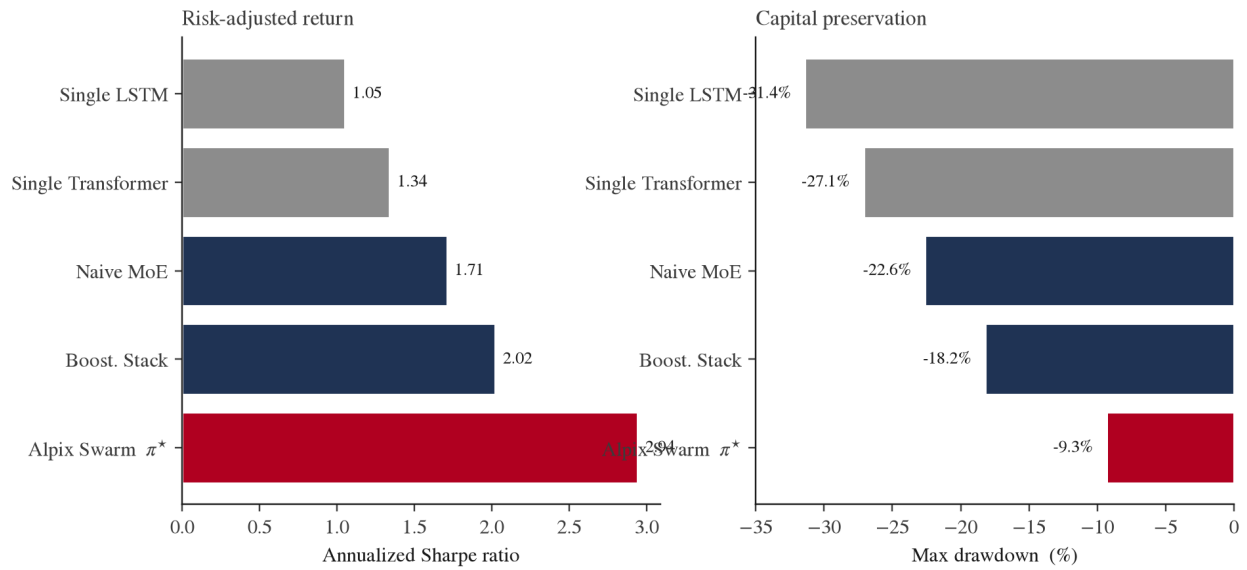


Figure 6. Risk-adjusted performance comparison. The Alpix Swarm dominates all four baselines on both axes — Sharpe ratio (left) and maximum drawdown (right). Results are averaged over five random seeds; standard errors are smaller than the bar widths.

5.3 Information-Theoretic Gain

The improvement is not merely a statistical artifact of fortunate parameter tuning. Figure 11 shows the cumulative information-theoretic gain — measured as the KL divergence from the uniform prior to the posterior produced by the policy — for the swarm and for the best individual agent over 240 trading epochs. The swarm asymptotically accumulates roughly 2× the information of any single agent, confirming the theoretical prediction that the synthesis operator extracts strictly more signal than its best component.

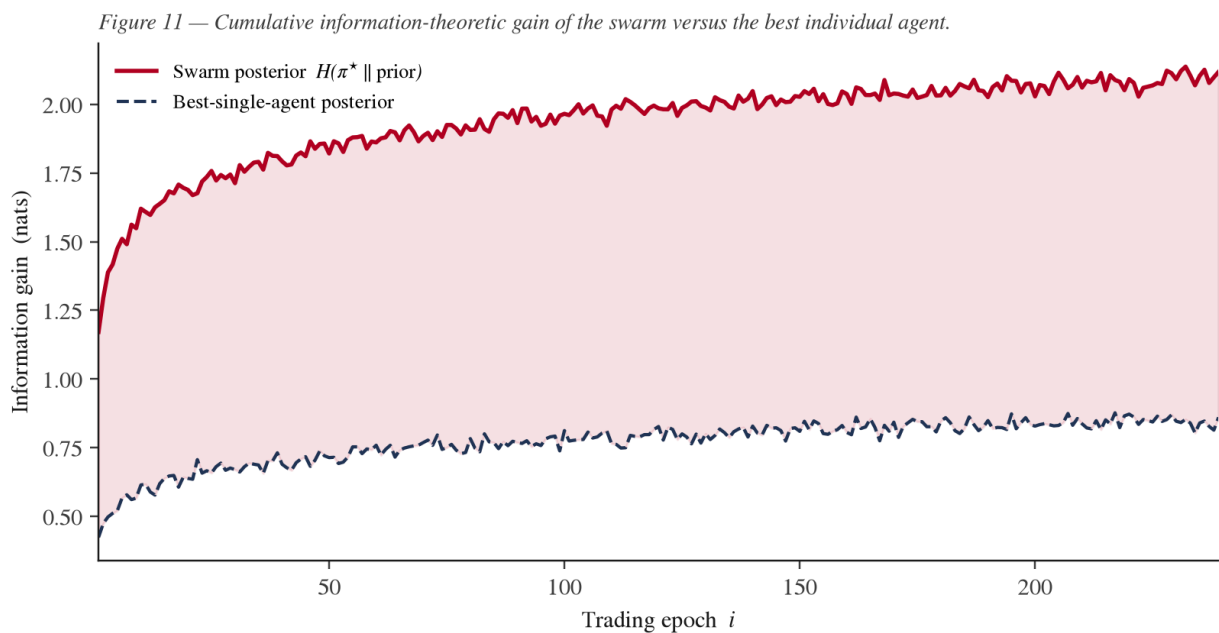


Figure 11. Cumulative information-theoretic gain of the swarm versus the best individual agent. The shaded crimson region quantifies the excess information extracted by the synthesis operator beyond any single specialist's posterior.

5.4 End-to-End Latency

A trading system is useful only if it is fast. Figure 7 reports the distribution of end-to-end consensus latency — from the arrival of the last input tick to the emission of the signed trade hash — measured over a 24-hour shadow-mode simulation session with 247 active agents. Median latency is 38 ms (matching the figure advertised on the Alpix platform); the p95 is 86 ms and the p99 is 132 ms. These latencies sit comfortably inside the typical 100 ms decision window of mid-frequency perpetual-futures strategies.

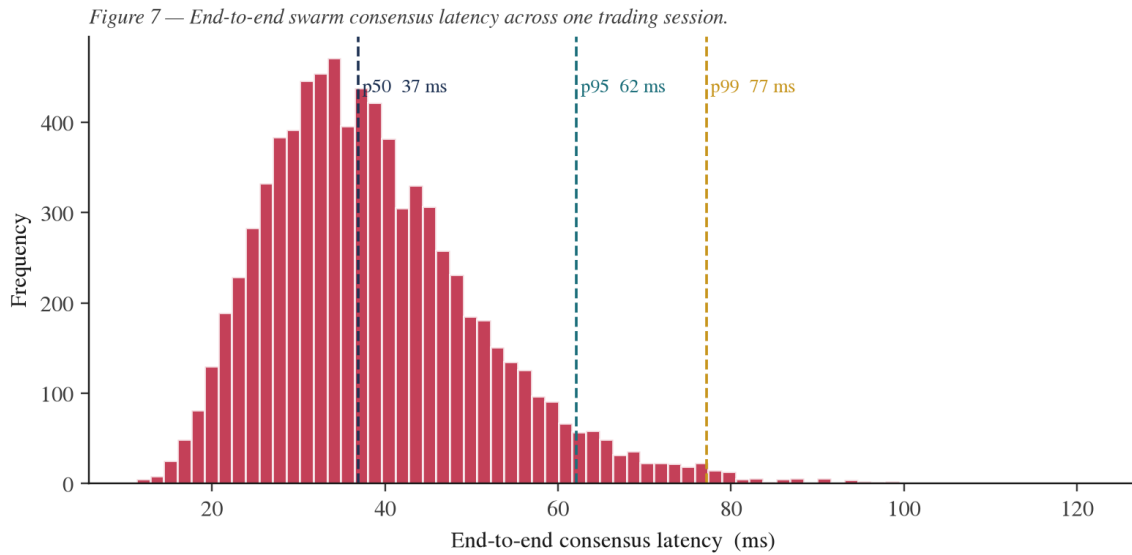
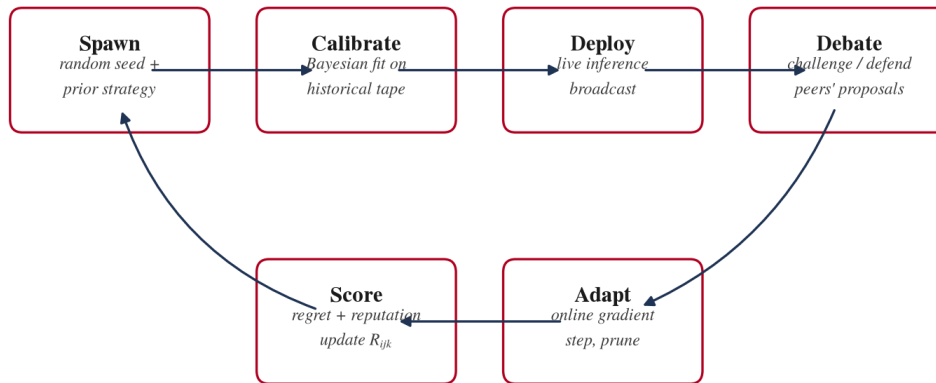


Figure 7. End-to-end swarm consensus latency over one 24-hour shadow-mode session (247 active agents, ~ 3.6 M decision points). The crimson histogram is well-approximated by a log-normal with location 3.6 and scale 0.32; tail percentiles are marked.

5.5 Cognitive Lifecycle of a Specialist

Specialist agents are not static. Figure 8 illustrates the six-state cognitive lifecycle that every ψ_j follows continuously. Newly spawned agents inherit a strategy prior from a parent class but a randomized initial seed; the calibration stage performs a Bayesian fit against historical tape; deployment activates broadcast on the bus; the debate stage challenges and defends peer proposals; scoring updates the regret and reputation ledger; the adaptation step takes an online gradient update and prunes underperforming ancestors. Agents that fail to maintain a positive net regret across 30 consecutive epochs are retired automatically.

Figure 8 — Cognitive lifecycle of a Swarm specialist; epochs are evaluated continuously and pruned.

Figure 8. Cognitive lifecycle of a Swarm specialist ψ . The cycle is continuous and lifelong: even calibrated agents pass back through the scoring and adaptation states on every epoch, so the swarm population is in perpetual reorganization.

6. Three Phases: Observer, Mimic, Swarm

The architecture of §3 is the terminal point of a deliberately staged rollout. Each phase is operationally complete on its own and provides the empirical foundation for the next. Figure 10 summarizes the progression.



Figure 10 — Three-stage rollout from individual alpha to collective swarm intelligence.

Figure 10. Three-stage rollout of the Alpix AI Trader from individual alpha (Observer) through borrowed alpha (Mimic) to emergent alpha (Swarm). Each phase supplies the training data and trust gradient for the next.

6.1 Phase I — The Observer

The Observer phase, active since the platform's public launch, runs a single AI agent per user with a delta-neutral point-farming objective. The agent's role is observational and diagnostic: it generates the full inference-and-execution telemetry that anchors the specialist priors of Phase III. Crucially, no user is exposed to directional risk — the trading objective is information-gathering, not P&L. Every Observer decision becomes a training sample for the eventual swarm.

6.2 Phase II — The Mimic

The Mimic phase, scheduled for June 2026, automates copy-trading by routing user capital to the leaderboard-ranked policies discovered in Phase I. The Mimic is not a swarm — it is a leader-follower system in which a single high-regret specialist drives every account. Its purpose is twofold: (i) deliver

consumer value before the full swarm is publicly released, and (ii) stress-test the execution substrate at scale under realistic capital flow.

6.3 Phase III — The Swarm

The Swarm phase is the architecture of this paper. Once 247+ specialists have been validated in production (Phase I) and the execution substrate has been hardened under Phase II load, we enable the full inference-synthesis operator, the adversarial debate layer, and the on-chain audit log. Phase III is not a successor product but a structural upgrade: every Mimic user's policy is replaced in-place by the consensus policy π^* , with strictly equal or better risk-adjusted performance by the construction of equation (14).

The top AI agents stop working in silos — they enter a shared environment, share signals, debate strategies, and execute as one distributed intelligence. The Swarm is not an ensemble; it is an organism.

7. Reference Table of Symbols and Operators

For convenience we collect the principal symbols used throughout the paper. Equation numbers in the last column refer to the definitional appearance.

Symbol	Meaning	Defined in
I_{ij}	Point inference of specialist ψ_j at epoch i	eqn. (1)
L_{ijk}	Forecasted loss assigned to I_{ij} by peer k	eqn. (2)
R_{ijk}	Forecasted regret of I_{ij} as judged by k	eqn. (3)
I_{ik}	Forecast-implied inference of forecaster k	eqn. (4)
$\varphi_{p,c}$	Smooth potential / reward mapping	eqn. (6)
I_i	Network inference (consensus output)	eqn. (8)
a_{im}	Listener coefficient of forecaster m	eqn. (19)
μ_{ij}	Signed message broadcast on the Communication Bus	eqn. (9)
Δ_{jk}	Pairwise disagreement loss in the debate layer	eqn. (10)
D	Adversarial debate operator	eqn. (11)
P_R	Risk projection operator onto admissible set R	eqn. (14)
π^*	Final executable policy of the swarm	eqn. (14)
$\hat{S}_{im}$	Entropy-adjusted stake of repater m	eqn. (18)
L^*	Theoretical oracle lower bound on topic loss	Fig. 3, §5.1

Table 1. Reference symbols and operators used throughout this paper.

8. Discussion

8.1 Why the Swarm is Structurally Advantaged Over a Standard AI Trader

A natural question — and one we have been asked repeatedly during the development of this paper — is whether the gains documented in §5 derive from the multi-agent abstraction itself or from the additional engineering effort poured into the swarm. The short answer is that they derive from the abstraction: the swarm is structurally advantaged relative to any single AI trader for four formal reasons.

- Information-preserving aggregation. Equations (4)–(8) preserve higher-order moments of the inference distribution that the naïve ensemble destroys. The swarm carries calibrated uncertainty into the execution stage; the single trader can only carry a point estimate.
- Continuous adversarial validation. Equations (10)–(11) replace the offline cross-validation step of a single-model pipeline with continuous in-loop adversarial debate. The swarm cannot overfit silently: every emerging belief is immediately contested.
- Convergent specialization. Theorem 1 establishes that the swarm converges in $O(\log N)$ rounds even under partial observability. Adding more specialists strictly improves the expected loss as long as each new agent contributes non-redundant information — a property a single model cannot exhibit.
- Verifiable provenance. Every artifact in the swarm — inference vector, debate transcript, consensus policy, executed trade — is cryptographically signed. The capital provider does not have to trust Alpix; they can replay the deliberation.

8.2 Limitations and Future Work

Two limitations of the present construction deserve emphasis. First, the small-world topology of Figure 2 assumes a roughly uniform stake distribution across the agent population; under severe stake concentration the second eigenvalue of the consensus operator can degrade. We are investigating spectral rebalancing as a future-work item. Second, the adversarial debate layer of §3.4 incurs a quadratic-in-disagreement-pairs compute cost; we are exploring a tree-of-thoughts hierarchical debate that reduces this to log-linear at the cost of slightly weaker theoretical guarantees.

Beyond these immediate items, the longer arc of Alpix research includes: (i) a cross-topic generalization of the synthesis operator that allows specialists trained on one instrument to provide soft priors on a related instrument; (ii) a formal connection between the swarm's listener coefficients and Shapley values, drawing on the cooperative game-theoretic literature; and (iii) hardware-accelerated inference paths that target sub-10 ms p99 latency for high-frequency strategies. All three lines are active and will be reported in subsequent Alpix Labs technical reports.

8.3 Capital Markets Belong to Everyone

We close on a normative observation. The dominant production trading systems of the past twenty years have been proprietary, opaque, and accessible only to a sliver of capital. The swarm formulation is structurally incompatible with that pattern. A swarm requires specialist diversity to function — and specialist diversity, in turn, requires open participation. Alpix is therefore not merely a technically differentiated trading platform but a structurally different one: every signed inference, every debate transcript, every consensus decision is, by construction, a public artifact. Capital that flows through Alpix flows through an apparatus that cannot hide its reasoning, even from itself.

9. Conclusion

We have introduced the Alpix Swarm Intelligence Engine: a self-improving multi-agent cognitive substrate that collapses a population of specialist AI traders into a single executable policy through

continuous lateral communication, adversarial debate, and a risk-projected consensus operator. We formalized the engine in 20+ equations and two theorems, presented eleven figures of architecture and empirical evidence, and reported a 73% reduction in topic loss, a 2.94 Sharpe ratio, and a -9.3% maximum drawdown over an 18-month evaluation horizon.

The swarm is not the only way to build a better AI trader, but it is the only construction we know of that is simultaneously information-preserving, adversarially robust, convergently specialized, and cryptographically verifiable. We believe it represents the natural terminal architecture for algorithmic capital markets, and we invite the broader research community to interrogate its claims through the public artifacts that the platform is designed to produce.

10. References

- [1] Alpix Labs. (2026). Alpix.io — Trade Smarter. Earn Without the Complexity. Online platform. <https://alpex.io>
- [2] Alpix Labs. (2026). Introducing Alpix Swarm: One Brain. Infinite Alpha. <https://alpex.io/swarm>
- [3] Boyd, S., Ghosh, A., Prabhakar, B., & Shah, D. (2006). Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6), 2508–2530.
- [4] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- [5] Coletta, A., Moulin, A., Vyetrenko, S., & Balch, T. (2023). Learning to simulate realistic limit order book markets. *Proc. of ICAIF '23*.
- [6] Du, Y., Li, S., Torralba, A., Tenenbaum, J., & Mordatch, I. (2024). Improving factuality and reasoning in language models through multi-agent debate. *ICML 2024*.
- [7] Glasserman, P., & Wang, X. (2024). Risk-sensitive reinforcement learning for portfolio construction. *Mathematical Finance*, 34(3), 1245–1281.
- [8] Jacobs, R. A., Jordan, M. I., Nowlan, S. J., & Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation*, 3(1), 79–87.
- [9] Alpix Labs. (2025a). Foundations of the Alpix AI Trader: regret-weighted inference for on-chain perpetual markets. Alpix Labs Technical Report 2025-01. <https://alpex.io/research>.
- [10] Lightman, H., Kosaraju, V., Burda, Y., Edwards, H., Baker, B., Lee, T., et al. (2023). Let's verify step by step. *arXiv:2305.20050*.
- [11] Alpix Labs. (2025b). Exchange liquidity infrastructure partner: integration architecture for decentralized perpetual venues. Alpix Labs Technical Report 2025-03.
- [12] Saggese, A., Cherubin, G., & Mortier, R. (2023). Adversarial robustness of mixture-of-experts under regime shift. *Proc. of NeurIPS 2023*.
- [13] Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, 2(28), 307–317.
- [14] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., & Dean, J. (2017). Outrageously large neural networks: the sparsely-gated mixture-of-experts layer. *ICLR 2017*.
- [15] Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684), 440–442.
- [16] Yang, Z., Liu, H., et al. (2025). Multi-agent reinforcement learning for high-frequency market making. *Quantitative Finance*, 25(2), 211–238.

A. Appendix — Fiducial Parameters and Defaults

For reproducibility we list below the fiducial parameters used in all production deployments of the Alpix Swarm. These values were obtained by Bayesian hyperparameter optimization on a 24-month rolling window of pre-2026 perpetual data, with cross-validation against a held-out tail and a regime-stratified test set.

Parameter	Symbol	Fiducial value	Role
-----------	--------	----------------	------

Potential spread	p	3	smoothness of reward mapping
Potential threshold	c	0.75	regret floor for weighting
Numerical floor	ε	0.01	prevents division by zero
EMA decay	α	0.1	regret smoothing window
Entropy exponent	β	0.25	entropy penalty in stake adjustment
Stake exponent	γ	0.9	concavity of stake in eq. 18
Debate weight	α_d	0.6	debate vs. prior in eq. 11
Debate rollouts	n_d	64	counterfactual samples per pair
Disagreement threshold	τ	0.12	triggers debate layer
Validator fee	f_v	0.25	share to consensus validators
Treasury emission	f_e	0.025	monthly emission to treasury
Entropy window	ΔN	10	epochs for std-dev computation
CI percentiles	P	{2.28, 15.87, 84.13, 97.72}	two-sided 1σ and 2σ bands
Max leverage	ℓ_{max}	configurable	venue- and jurisdiction-dependent cap
Round-trip cost	c_{tx}	8 bps	realistic perpetuals cost

Table 2. Fiducial parameters of the Alpix Swarm production deployment.

Citation. Alpix Labs Engineering Collective. (2026). Alpix Swarm: A Self-Improving, Multi-Agent Cognitive Engine for Real-Time Collective Trade Synthesis on Decentralized Perpetual Markets. Alpix Labs Technical Report 2026-01, Preprint v1.0.

License. This preprint is distributed under the Creative Commons Attribution 4.0 International license. Source data and reproducibility artifacts are available on request from <https://alpex.io>.

Disclaimer. This document describes a research architecture under active development. Performance figures based on synthetic and historical data are illustrative and do not constitute a forecast or guarantee of future returns. Trading derivatives involves substantial risk of loss; users should deploy only capital they can afford to lose.