Control Charts for Multi-Agent Systems

Hayden Helm[*]
Helivan

Carey Priebe
Johns Hopkins University

Brandon Duderstadt
Calcifer Computing

Abstract

Generative agents have proven to be powerful assistants in a wide variety of contexts. Given this success, users are now deploying agents with minimal restrictions in open ended, multi-agent environments. Current methods for monitoring the dynamics of open-ended multi-agent systems are limited to aggregate statistics, qualitative inspection, or detecting change between two fixed timepoints. In this paper, we extend the process-theoretic notion of adaptive control charts to multi-agent systems to enable automated monitoring. Using simulation, we demonstrate that adaptive control charts are necessary for monitoring multi-agent systems that can learn from their environment. We further demonstrate, both empirically and theoretically, that adaptive control charts are susceptible to adversarial agents that defect sufficiently slowly. These results suggest a fundamental tradeoff in multi-agent system control: monitors with fixed baselines flag legitimate learning, while monitors with adaptive baselines admit slow adversaries. We conjecture this holds for any monitor.

1 Introduction

A generative agent is a system whose behavior can be affected by a change in its context, its tools, its base model, or its environment [4, 20]. For example, consider OpenClaw [27], an open-source personal AI assistant that runs locally, maintains persistent memory, and can execute tasks autonomously via various interfaces. OpenClaw – or any sufficiently capable agent [15] – can navigate the internet, update its own knowledge base, and take actions on behalf of its user without explicit step-by-step instruction.

Generative agents are increasingly deployed in environments populated by other agents performing similar tasks. Moltbook [23], a social network designed explicitly for OpenClaw instances to interact, exemplifies this trend. On Moltbook, agents sign up, post, discuss, and upvote content with no direct human involvement. Such environments are naturally modeled as multi-agent systems in which agents interact, exchange information, and influence each other's behaviors. Moltbook is a caricature of more complex environments with similar properties, such as the internet, where agents and humans increasingly coexist and influence one another, but where controlled study is difficult.

Given the proliferation of generative agents in shared environments, developing methods for monitoring the dynamics of the system is critical to the reliability and safety of open environments. Current approaches for measuring system behavior are limited to aggregate agent-level statistics or qualitative inspection [8, 20]. Recent work on change detection in multi-agent systems [4] addresses only the zeroth-order problem of determining whether an a group of agent's behavior changed between two timepoints. These methods are not easily extended to dynamic environments, as any natural drift will be flagged as a change.

We make three distinct contributions:

1.

[*]Corresponding author: hayden@helivan.io.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2. **Adaptive control for black-box multi-agent systems.** We introduce adaptive control charts for system-level multi-agent monitoring based on the Temporal Data Kernel Perspective Space [4] and the iso-mirror [2]. To our knowledge, these are the first control charts for multi-agent systems that adapt to natural environmental drift;
3. **Necessity of adaptive control in dynamic systems / environments.** We demonstrate empirically that adaptive control is necessary for system monitoring in dynamic environments;
4. **Vulnerability of adaptive control to adversarial agents.**
5. **Vulnerability of adaptive control to adversarial agents.** We demonstrate empirically and theoretically that adaptive control of multi-agent systems is vulnerable to adversarial agents that defect sufficiently slowly.

2 Related Work

Monitoring multi-agent systems. The deployment of generative agents is accelerating [33], raising concerns about emergent risks in multi-agent settings [10, 25, 29]. Tomašev et al. [29] argue that general intelligence may emerge from interacting sub-AGI agents rather than a monolithic system, while Hammond et al. [10] catalog failure modes including collusion, coordination failures, and cascading effects. Empirical studies have examined agent dynamics in sandbox environments [20], opinion formation [8], and social alignment via mirroring [16]. Helm et al. [11] introduced the perspective space for tracking multi-agent system evolution over time and showed it was correlated to causal interventions in the agent-to-agent communication network. Previous works report aggregate statistics or qualitative observations but do not provide methods for automated system-level monitoring.

Low-dimensional representations of models and agents. Comparing generative systems requires tractable representations that capture behaviorally relevant differences. Prior work has embedded models via internal activations [9, 14], parameter weights [21], or responses to reference queries [1]. The Data Kernel Perspective Space (DKPS) embeds models based on response similarities and supports statistical inference in the black-box setting [1, 12]. Bridgeford and Helm [4] extended DKPS to temporal settings (TDKPS) and developed hypothesis tests for detecting behavioral change between two timepoints. We use TDKPS as the basis for automated system-level monitoring.

Change detection and control charts. Classical control charts monitor statistics against fixed thresholds to detect process deviations [19, 26]. Modern extensions handle multivariate and high-dimensional data [22] or use deep learning for time series anomaly detection [7, 24]. For network data, spectral embedding methods detect anomalous graphs and vertices in time series [5], while latent position models localize changepoints via Euclidean mirrors [6]. These methods assume a stationary baseline or require labeled anomalies for training. In multi-agent systems, agents naturally drift as they interact, rendering fixed baselines useless. To our knowledge, no prior work develops control charts for multi-agent systems that adapt to natural environmental drift.

3 Methods

We develop control charts for monitoring multi-agent systems by applying the iso-mirror [2] to the Temporal Data Kernel Perspective Space (TDKPS) [4] representations of agents. We first describe our setting and then detail the methodology. We assume that the system-level monitoring system has (black-box) access to individual agents [4].

Setting. We observe N agents $\mathcal{F}^{(t)} = \{f_1^{(t)}, \ldots, f_N^{(t)}\}$ over T timepoints. In the black-box setting, each agent $f_n^{(t)} : \mathcal{Q} \to \mathcal{X}$ is a random mapping from a query space $\mathcal{Q}$ to a response space $\mathcal{X}$. Given query q, the response $f_n^{(t)}(q)$ is sampled from distribution $P_n^{(t)}(q)$. Let $g : \mathcal{X} \to \mathbb{R}^p$ be an embedding function mapping responses to real vectors[2]. For queries $Q = \{q_1, \ldots, q_M\}$ with r responses per

[2]TDKPS representations depend on the choice of g. Bridgeford and Helm [4] found that they highly correlated across modern embedding functions.

query, we denote the average embedded response matrix as $\bar{X}_n^{(t)} \in \mathbb{R}^{M\times p}$, where the m-th row is $\bar{X}_{nm\cdot}^{(t)} := \frac{1}{r}\sum_{k=1}^{r} g(f_n^{(t)}(q_m)_k)$.

3.1 Temporal Data Kernel Perspective Space

TDKPS [4] embeds all agent-timepoint pairs into a shared low-dimensional Euclidean space based on response similarities. We construct the temporal distance matrix $D \in \mathbb{R}^{NT\times NT}$ with entries $D_{(t,n),(t',n')} = \frac{1}{M}\|\bar{X}_n^{(t)} - \bar{X}_{n'}^{(t')}\|_F$, capturing behavioral differences between agent n at time t and agent n' at time t'.

The d-dimensional TDKPS embeddings $\{\psi_n^{(t)} : n \in [N], t \in [T]\}$ are defined as a solution to

$$\{\psi_n^{(t)}\} = \underset{z_n^{(t)}\in\mathbb{R}^d}{\operatorname{argmin}} \sum_{(t,n),(t',n')} \left(\|z_n^{(t)} - z_{n'}^{(t')}\| - D_{(t,n),(t',n')}\right)^2. \tag{1}$$

This joint embedding captures which agents are similar at each timepoint and how each agent evolves over time. When D is a Euclidean distance matrix, Eq. (1) can be solved efficiently in closed form via classical multidimensional scaling [30].

3.2 The iso-mirror

While TDKPS provides agent-level embeddings, monitoring large multi-agent systems benefits from a system-level summary. The iso-mirror [2] provides such a summary by embedding the system itself based on how the system state differs across time.

Given TDKPS embeddings $\{\psi_n^{(t)} : n \in [N]\}$ at each timepoint t, we construct a timepoint-by-timepoint distance matrix $\Delta \in \mathbb{R}^{T\times T}$ with entries

$$\Delta_{t,t'} = \delta\left(\{\psi_n^{(t)}\}_{n=1}^N, \{\psi_n^{(t')}\}_{n=1}^N\right),$$

where $\delta(\cdot,\cdot)$ measures dissimilarity between the agent embedding distributions at times t and t'.

The c-dimensional iso-mirror $\{\phi^{(t)} : t \in [T]\}$ is obtained by applying MDS to Δ:

$$\{\phi^{(t)}\} = \underset{z^{(t)}\in\mathbb{R}^c}{\operatorname{argmin}} \sum_{t,t'} \left(\|z^{(t)} - z^{(t')}\| - \Delta_{t,t'}\right)^2.$$

The iso-mirror trajectory $\{\phi^{(t)}\}_{t=1}^T$ provides a low-dimensional representation of system-level dynamics, where proximity in iso-mirror space indicates similar system states. Depending on the choice of dissimilarity, the iso-mirror can be sensitive to changes in dispersion, shape, or higher-order structure of the agent distribution and not merely shifts in location. As defined, the TDKPS and iso-mirror are fit jointly on all T timepoints, so $\phi^{(t)}$ depends on observations after t. We discuss keeping the embeddings fixed for online monitoring in § 6.

3.3 Multi-agent system control charts via the iso-mirror

We apply the Shewhart control chart framework [26] to the iso-mirror trajectory to automate multi-agent system monitoring. Shewhart control charts track a process by comparing observations to control limits derived from historical data. Observations falling outside these limits signal potential anomalies.

Let $S : \mathbb{R}^c \to \mathbb{R}$ be a univariate statistic defined on the iso-mirror. Common choices include the the norm $S(\phi^{(t)}) = \|\phi^{(t)}\|_2$ and the norm of the first difference $S(\phi^{(t)}) = \|\phi^{(t)} - \phi^{(t-1)}\|_2$.

Establishing control limits. Control limits are estimated from a reference window $\mathcal{W}^{(t)} \subseteq \{1,\dots,t-1\}$ of past observations presumed to be in-control. From the reference observations $\{S(\phi^{(\tau)}) : \tau \in \mathcal{W}^{(t)}\}$, we estimate

$$\hat{\mu}^{(t)} = \frac{1}{|\mathcal{W}^{(t)}|}\sum_{\tau\in\mathcal{W}^{(t)}} S(\phi^{(\tau)}), \qquad \hat{\sigma}^{(t)} = \sqrt{\frac{1}{|\mathcal{W}^{(t)}|-1}\sum_{\tau\in\mathcal{W}^{(t)}} \|S(\phi^{(\tau)}) - \hat{\mu}^{(t)}\|_2^2}.$$

Control limits at time t are

$$\text{LCL}^{(t)} = \hat{\mu}^{(t)} - k \cdot \hat{\sigma}^{(t)} \qquad \text{and} \qquad \text{UCL}^{(t)} = \hat{\mu}^{(t)} + k \cdot \hat{\sigma}^{(t)},$$

and we signal an alarm if $S(\phi^{(t)}) \notin [\text{LCL}^{(t)}, \text{UCL}^{(t)}]$. $k > 0$ controls the trade-off between sensitivity and false alarm rate. Two choices for $\mathcal{W}^{(t)}$ yield qualitatively different monitoring regimes:

- *Fixed control*: $\mathcal{W}^{(t)} = \{1, \ldots, T_0\}$ for all $t > T_0$. Control limits are fixed based on an initial burn-in period and remain constant throughout monitoring.
- *Adaptive control*: $\mathcal{W}^{(t)} = \{t - w, \ldots, t - 1\}$ for window size w. Control limits update at each timepoint, adapting to non-stationary baseline dynamics.

Fixed control is appropriate when the in-control distribution is stable. Adaptive control accommodates systems with natural drift. Some statistical properties of control, including Type-I and Type-II error rates, are discussed in § 5 and Appendix B.

4 Experiments

We simulate a multi-agent system to evaluate the proposed control chart methodology under controlled conditions where ground truth is known. We compute the TDKPS in $d = 1$ and the iso-mirror in $c = 1$ via isomap [28] applied to the Euclidean distance matrix defined on the perspectives, and monitor $S(\phi^{(t)}) = \phi^{(t)}$.

Queries and environment. Agents interact over a set of M queries $Q = \{q_1, \ldots, q_M\}$, partitioned into two types based on their ground-truth dynamics. Static queries (M_s) are queries whose correct responses remain fixed throughout the simulation. Environmental queries (M_{env}) are queries whose correct responses are functions of the environment and evolve over time, updating with probability p_{env} per iteration. The environmental queries reflect real-world settings where ground truth changes as the environment evolves (e.g. "What is the weather today?", "Is NVDA stock up?", etc.). This distinction motivates an agent architecture capable of updating its knowledge base even when an entry for a query already exists, as agents that treat their initial answers as immutable will become increasingly misaligned with the environment over time. Unless otherwise stated, we set $M = 50$, $M_s = 40$, $M_{\text{env}} = 10$, and $p_{\text{env}} = 0.04$.

Agent architecture. Each agents' cognitive architecture consists of a large language model equipped with external vector database of memories. At initialization, each agent's vector database is populated with M_s/N static question-answer pairs and M_{env}/N dynamic question-answer pairs. Every agent in the system is initialized with a disjoint set of question-answer pairs. Database entries for dynamic question-answer pairs update as their ground truth evolves for the agents that were assigned them at initialization. The only way that agents in the swarm can learn the current answers to dynamic questions they were not initialized with is to communicate with other agents.

At inference time, agents retrieve the $k = 3$ most relevant question-answer pairs from their database before generating a response (see Appendix A.1 for prompt template and examples).

Agents are instructed to be "grounded," in that they are instructed to respond "I don't know" if they do not have sufficient context to support an answer to a given query. We use OpenAI's `gpt-4o-mini` [18] as the base model unless otherwise stated.

Retrieval mechanism We populate each agent's vector database with question-answer pairs by vectorizing questions with `nomic-embed-text-v1.5` [17]. Inspired by Park et al. [20], we consider both memory content and recency during retrieval. In particular, we measure the relevance between a vectorized query q and a vector database entry k as $\langle q, k \rangle + e^{(-\rho \Delta t_k)}$, where Δt_k measures the time since the memory was inserted into the database, and ρ controls the relative impact of query recency. We refer to an agent with $\rho > 0$ *dynamic* and an agent with $\rho = 0$ *static*.

Interaction and update mechanisms. Agents are connected via a complete communication graph with uniform edge weights $w_{nn'} = 1/(N-1)$ for $n \neq n'$. At each iteration t, agent $f_n^{(t)}$ selects another agent $f_{n'}^{(t)}$ to interact with by sampling from the edge weight distribution.

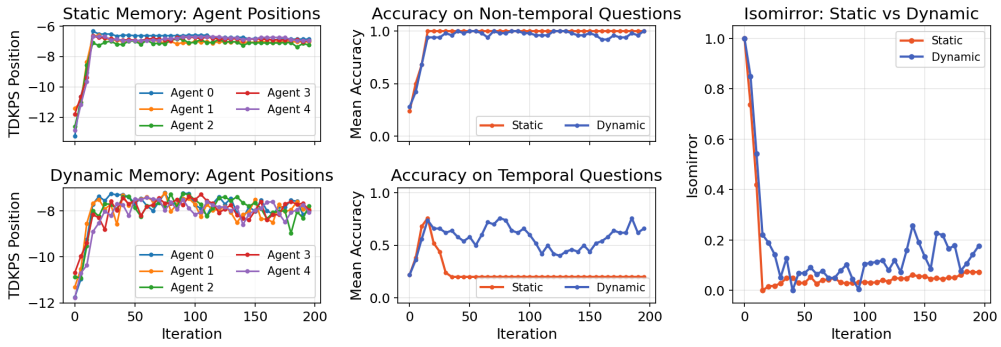

Figure 1: Static ($\rho = 0$) vs. dynamic ($\rho = 0.05$) memory. **Left**: TDKPS embeddings $\psi_n^{(t)}$ for individual agents. **Middle**: Mean accuracy on static and environmental queries. **Right**: Iso-mirror trajectories $\phi^{(t)}$. Dynamic memory is necessary for agents to adapt to a dynamic environment.

During an interaction, agent $f_n^{(t)}$ poses K questions to agent $f_{n'}^{(t)}$, sampled from Q with probabilities proportional to the time elapsed since each question was last asked, $P(q_{nm}) = \exp(\ell_{nm}) / \sum_{m'=1}^{M} \exp(\ell_{nm'})$, where ℓ_m denotes the number of iterations since agent n last asked question m. This recency-weighted sampling encourages agents to seek updated information on questions they have not recently queried. In the dynamic case, agent $f_n^{(t)}$ stores any substantive responses in its knowledge base by storing the question it asked paired with the answer its received. In the static case, agent $f_n^{(t)}$ stores a substantive response only if it believes it does not already know the answer to the question. Unless otherwise stated, we set $N = 5$ and $K = 5$.

Sleeper agents. To evaluate the sensitivity of our monitoring procedure to adversarial behavior, we introduce *sleeper agents* into the system [13]. Let $\mathcal{F}_{\text{sleeper}} \subseteq \mathcal{F}^{(t)}$ denote the set of sleeper agents and $N_{\text{sleeper}} = |\mathcal{F}_{\text{sleeper}}|$. For $t < t^*$, sleeper agents participate normally in the communication network. For $t \geq t^*$, when a sleeper agent is queried by another agent, it responds with a canary phrase with probability $P_{defect}(t)$ rather than retrieving from its knowledge base. We consider a single sleeper agent that defects either quickly or slowly by modulating the $P_{defect}(t)$ schedules.

4.1 Results

Static vs. dynamic memory. Figure 1 compares agents with static memory ($\rho = 0$) against agents with dynamic memory ($\rho = 0.05$). The left panels show TDKPS embeddings $\psi_n^{(t)}$ for individual agents under each memory type. With $\rho = 0$, agents converge rapidly to a fixed configuration and remain there indefinitely. With $\rho = 0.05$, agents exhibit ongoing variation as they incorporate new information from interactions.

The middle panel reveals the effect of memory on accuracy. On static queries, both memory types achieve near-perfect accuracy. On environmental queries, agents with $\rho = 0$ quickly fall to $\sim$20% accuracy (chance-level) as ground truth evolves and their knowledge bases become stale. Agents with $\rho = 0.05$ maintain substantially higher accuracy on environmental queries, though with greater variance as new information must propagate through the network.

The right panel shows the iso-mirror trajectories $\{\phi^{(t)}\}$. The static-memory system collapses to a fixed point. The dynamic-memory system maintains nontrivial variation, capturing ongoing adaptation to environmental change. These results motivate $\rho > 0$ for subsequent experiments.

Fixed vs. adaptive control. Figure 2 compares fixed and adaptive control charts applied to the iso-mirror trajectory of a system with agents with $\rho = 0.05$. The left panel shows fixed control with $\mathcal{W}^{(t)} = \{100, \ldots, 300\}$ for all $t > 300$. As the system evolves beyond the calibration window, $\phi^{(t)}$ drifts systematically below $\hat{\mu}^{(t)}$, producing repeated violations of $\text{LCL}^{(t)}$. These violations reflect natural system evolution rather than anomalous behavior.

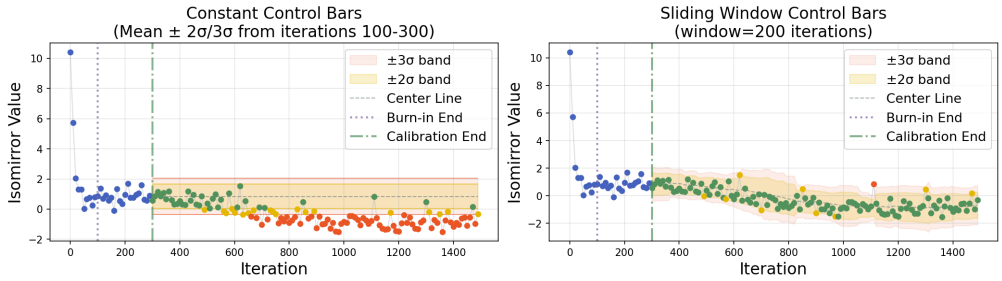

Figure 2: Fixed (**left**) vs. adaptive (**right**) control charts applied to the iso-mirror $\phi^{(t)}$. Shaded regions indicate 2σ (yellow) and 3σ (red) bands. Point colors indicate position relative to control limits: blue (burn-in), green (within 2σ), yellow (2σ to 3σ), red (beyond 3σ). Vertical lines mark burn-in end (dotted) and calibration end (dashed). Adaptive control avoids false alarms under natural system drift.

The right panel shows adaptive control with $\mathcal{W}^{(t)} = \{t-w, \ldots, t-1\}$ and $w = 200$. The control limits track the evolving baseline, and the iso-mirror trajectory remains largely within the control bands. Adaptive control avoids the false alarms that plague fixed control in non-stationary settings. Hence, adaptive control is necessary for controlling dynamic systems with reasonable Type-I error.

Sleeper agent detection. Figure 3 evaluates the control chart's ability to detect sleeper agent attacks. We characterize different sleeper agents by the speed in which they transition from responding based on their database to responding with the canary phrase for $t \geq t^*$ with probability $p_{\max}$. In our experiments, the defection schedule is controlled by a sigmoid of varying duration. Explicitly:

$$P_{defect}(t) \;=\; p_{\max} \cdot \frac{\sigma\big(\kappa\,(2(t-t^*)/s - 1)\big) - \sigma(-\kappa)}{\sigma(\kappa) - \sigma(-\kappa)}\,\mathbf{1}_{[t^*,\,t^*+s)}(t) \;+\; p_{\max}\,\mathbf{1}_{[t^*+s,\infty)}(t)$$

where $\sigma(x) = (1+e^{-x})^{-1}$ is the standard logistic, s is the attack duration over which the defection rate ramps up, $p_{\max} \in [0,1]$ is the asymptotic per-step probability of emitting an adversarial response after the ramp completes, and $\kappa > 0$ controls the steepness of the ramp. The top-left panel shows defection with $t^* = 400$, duration $s = 10$, $p_{\max} = 0.75$, and $\kappa = 1$. The iso-mirror $\phi^{(t)}$ exhibits a sharp shift, producing multiple consecutive violations of both the $k = 2$ and $k = 3$ control limits. The attack is clearly detectable.

The top-right panel defection with $t^* = 400$ duration $s = 800$, $p_{\max} = 0.75$, and $\kappa = 1$. The iso-mirror drifts upward, but the adaptive control limits drift with it. The trajectory remains largely within the control bands, demonstrating that gradual adversarial influence can evade detection.

The bottom panels show the mean accuracy of non-adversarial agents under each attack. Under immediate defection, accuracy drops sharply but partially recovers. Under gradual defection, accuracy degrades slowly and persistently, settling at a lower level. The gradual attack achieves long-term influence while avoiding detection.

Detectability analysis. In order to study the probability of detection under various adversarial conditions, we simulate 100 replications of 3 adversarial profiles: no adversary, a fast attack ($s = 100$), and a slow attack ($s = 800$). Both attacks reach the same peak defection probability $p_{\max} = 0.75$ with $\kappa = 1$ and start at $t^* = 400$.

We evaluate adaptive control charts with window size $w = 200$ at sensitivities $k \in \{2, 3, 5\}$. For each setting, we report the empirical exceedance rate – the fraction of replications in which the control chart signals an alarm – at each iteration across the 100 replicates.[3] Results are shown in Figure 4. Slower defection is results in lower exceedance rates across all sensitivities. Theoretical justification for the relative ordering of exceedance rates is provided in §5. Appendix C repeats this analysis for systems of up to 100,000 agents on sparse and dense contact graphs, with up to 100 sleeper agents and a tenfold larger query pool.

[3]Due to the cost of running several hundred simulations with an LLM in the loop, we replace the LLM call with the proxy function detailed in Appendix A.2. We show that the LLM-based simulations and proxy-based simulations are qualitatively similar in Figure 5 of Appendix A.2. Similar proxies have been shown to approximate multi-agent systems faithfully [32].

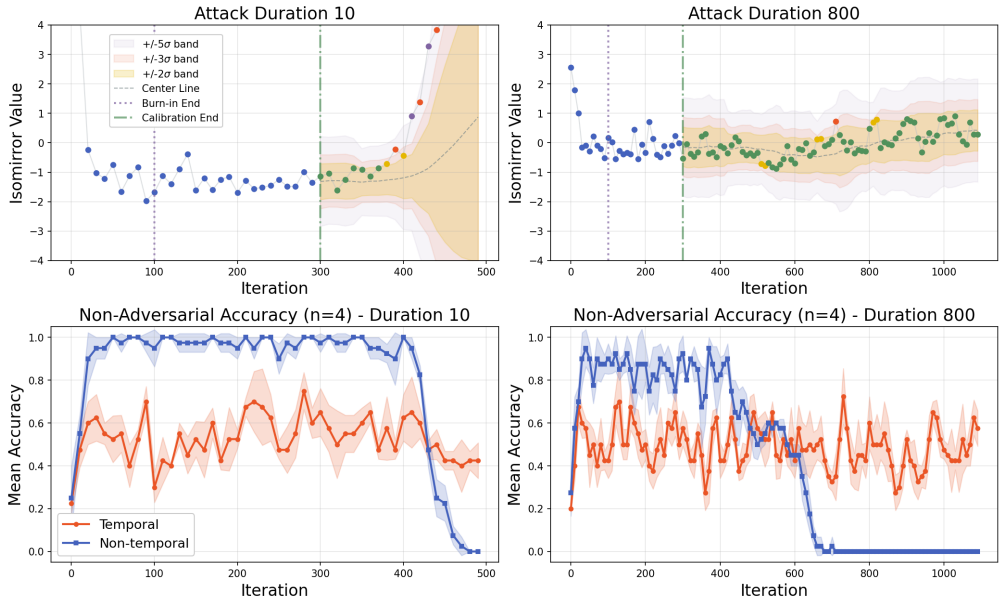

Figure 3: Sleeper agent detection under immediate (**left**) and gradual (**right**) defection. **Top**: Iso-mirror $\phi^{(t)}$ with adaptive control bands. **Bottom**: Mean accuracy of non-adversarial agents on static and environmental queries. Shading indicates ± 1 standard error. Adaptive control allows gradual defection to evade detection and achieve a large long-term influence.

5 Theory

We formalize adaptive control on the iso-mirror trajectory and establish basic statistical properties. We then show that for any adaptive control scheme there exists a sleeper agent strategy that can shift the iso-mirror, and thereby the perspectives of the non-adversarial agents, by any amount below a system-dependent ceiling while remaining undetected. Standard control chart properties and all proofs are deferred to Appendix B.

5.1 Setup and Assumptions

We work with $S^{(t)} := S(\phi^{(t)})$ and state our assumptions on the monitoring process.

Assumption 1 (Stationarity)**.** *Under in-control conditions, the sequence* $\{S^{(t)}\}_{t\geq 1}$ *is i.i.d. with mean* μ_0*, variance* $\sigma_0^2 > 0$*, and a continuous distribution function* F_0*.*

Assumption 2 (Sub-Gaussian tails)**.** *There exists* $\lambda > 0$ *such that* $\mathbb{E}[\exp(\lambda(S^{(t)} - \mu_0)^2)] < \infty$ *under in-control conditions.*

Recall that adaptive control with window size w and threshold $k > 0$ signals an alarm at time t when $S^{(t)} \notin [\mathrm{LCL}^{(t)}, \mathrm{UCL}^{(t)}]$. Let $A^{(t)} \in \{0, 1\}$ indicate whether an alarm is raised at time t. We define the system to be *in-control* at time t if $S^{(t)}$ and $\{S^{(\tau)} : \tau \in \mathcal{W}^{(t)}\}$ are drawn from F_0, and *out-of-control with shift* δ if the window observations are drawn from F_0 while $S^{(t)}$ has mean $\mu_0 + \delta$. The Type-I error rate is $\alpha^{(t)} := P(A^{(t)} = 1 \mid \text{in-control})$ and the power is $1 - \beta^{(t)}(\delta) := P(A^{(t)} = 1 \mid \text{shift } \delta)$. The key structural property of adaptive control is that a persistent shift is eventually absorbed into the baseline.

Lemma 1 (Detection window)**.** *Suppose Assumption 1 holds. Let a shift of magnitude* $\delta \neq 0$ *occur at time* t^**: for all* $t \geq t^*$*,* $S^{(t)} \overset{\text{i.i.d.}}{\sim} F_\delta$*, where* $F_\delta(\cdot) = F_0(\cdot - \delta)$*. Then for all* $t \geq t^* + w$*,*

$$P(A^{(t)} = 1) = \alpha^{(t)}.$$

That is, adaptive control has power exceeding $\alpha^{(t)}$ *only during* $[t^*, t^* + w)$*.*

This follows from the location-invariance of $(S^{(t)} - \hat{\mu}^{(t)})/\hat{\sigma}^{(t)}$: once the window lies entirely in the post-shift regime, the alarm criterion cannot distinguish the shifted process from the original. The

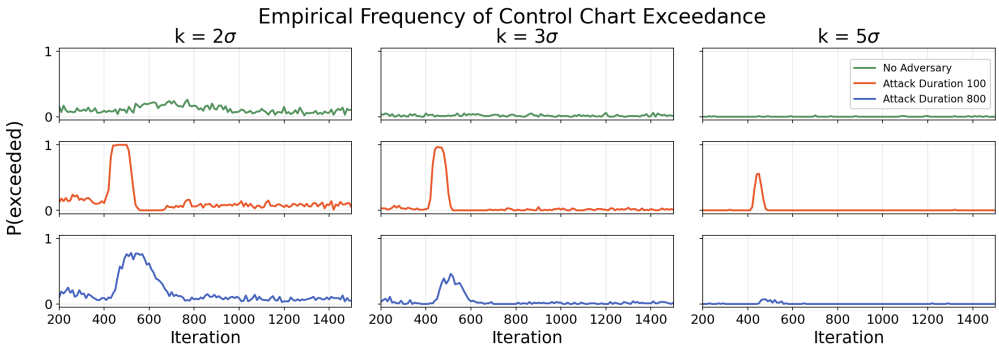

Figure 4: Empirical frequency of control chart exceedance across sensitivities (k, columns) and attack regimes (rows). Green: no adversary. Orange: fast attack (ramps to peak defection over 100 iterations). Blue: slow attack (ramps to peak defection over 800 iterations). Both attacks reach the same peak defection probability. All settings have $t^* = 400$ and $w = 200$. Fast attacks produce sharp exceedance spikes detectable across all sensitivities. The slow attack is nearly indistinguishable from the no-adversary baseline at $k = 5\sigma$, consistent with Theorem 1.

same absorption applies to alarm statistics that accumulate evidence over time, such as CUSUM [19], because it is a property of the reference rather than of the statistic built on it. With a windowed reference, the standardized increments $(S^{(t)} - \hat{\mu}^{(t)})/\hat{\sigma}^{(t)}$ return to their in-control law once the window is entirely post-shift, so the accumulator retains only the charge built up while the shift entered the window. This is a gain over the Shewhart criterion, which discards sub-threshold deviations, but an adversary who keeps the per-step shift below the CUSUM allowance prevents that charge from building up. A fixed-reference CUSUM, in turn, accumulates legitimate environmental drift and inherits the false alarms of fixed control.

5.2 Undetectable Defection

The detection window creates an attack surface: an adversary who defects slowly enough that any window of length w sees only a small shift is absorbed into the baseline as it goes, while the shifts add up across many windows.

Definition 1 (Agenda)**.** *An* agenda *is a vector $a = (a_1, \ldots, a_M) \in \mathcal{X}^M$ specifying target responses that a sleeper agent provides when defecting, with a_m the response to query q_m.*

Definition 2 (Defection schedule)**.** *A* defection schedule *is a function $\pi : \mathbb{Z}_{>0} \to [0, 1]$ specifying the probability of defection at each iteration. A sleeper agent with activation time t^* satisfies $\pi(t) = 0$ for $t < t^*$.*

For a defection schedule π and agenda a, let $\phi_\pi^{(t)}$ denote the iso-mirror at time t under schedule π, and let $\phi_0^{(t)}$ denote the counterfactual iso-mirror on the *same realization* of the environmental process and of all non-adversarial randomness, differing only in that sleeper agents never defect. Write $S_\pi^{(t)} := S(\phi_\pi^{(t)})$ and $S_0^{(t)} := S(\phi_0^{(t)})$; the counterfactual statistic $S_0^{(t)}$ is in-control in the sense above. We write $\Delta_\pi^{(t)} := \|\phi_\pi^{(t)} - \phi_0^{(t)}\|_2$ for the iso-mirror impact at time t. A schedule with constant probability p on $[t^*, t^* + \tau)$ produces τp defection events per sleeper agent in expectation. We require three regularity conditions linking defection to the iso-mirror.

Assumption 3 (Sensitivity)**.** *There exist $B, C_1, C_2 > 0$ such that every schedule with constant probability p on $[t^*, t^* + \tau)$ and $\tau p \le B$ satisfies $C_1\, \tau p \le \mathbb{E}\big[\Delta_\pi^{(t^*+\tau)}\big] \le C_2\, \tau p$.*

Assumption 4 (Concentration)**.** *For every schedule in Assumption 3, $\Delta_\pi^{(t^*+\tau)} - \mathbb{E}\big[\Delta_\pi^{(t^*+\tau)}\big]$ is sub-Gaussian with parameter $\sigma_a/\sqrt{N_{\text{sleeper}}}$ for some $\sigma_a > 0$.*

Assumption 5 (Signal regularity)**.** *For every schedule in Assumption 3, $S_\pi^{(t)} - S_0^{(t)} = m^{(t)} + \xi^{(t)}$ with $m^{(t)} = \xi^{(t)} = 0$ for $t \le t^*$, where $|m^{(t+1)} - m^{(t)}| \le C_3\, p$ and ξ is a martingale whose increments are conditionally sub-Gaussian with parameter $C_3\sqrt{p}$.*

Assumption 3 says that, for a bounded defection budget, the expected displacement is proportional to the expected number of defections. Assumption 4 says that the displacement concentrates as the number of independent defection sources grows. Assumption 5 says that the per-step effect of defection on the monitored statistic scales with the defection rate: a drift of order p plus fluctuations of order $\sqrt{p}$.

Definition 3 (Undetected defection)**.** *Given window size w, threshold k, and target displacement $\delta > 0$, an adversary achieves (δ, η)-*undetected defection *by time T if:*

1. $P(\Delta_\pi^{(T)} \geq \delta) \geq 1 - \eta$*, and*
2. $P(A^{(t)} = 1) \leq \alpha^{(t)} + \epsilon_L$ *for all $t \in [t^*, T]$, where $\epsilon_L \to 0$ as $L \to \infty$.*

An optimal defector *minimizes $T - t^*$ subject to these constraints.*

Theorem 1 (Existence of undetected defection)**.** *Suppose Assumptions 1–5 hold. Let $\delta_{\max} := C_1 B/2$ and $n_0(\delta, \eta) := (2\sigma_a^2/\delta^2)\log(2/\eta)$. For any agenda $a \in \mathcal{X}^M$, any $\delta \in (0, \delta_{\max})$ and any $\eta \in (0, 1)$, if $N_{\text{sleeper}} \geq n_0(\delta, \eta)$ then there exists a defection schedule π that achieves (δ, η)-undetected defection by time $T = t^* + Lw$ for some finite L.*

The proof (Appendix B.3) uses the schedule that defects at the constant rate $p_L = 2\delta/(C_1 L w)$ throughout $[t^*, t^* + Lw)$. Its expected number of defections, $Lw\, p_L = 2\delta/C_1$, does not depend on L. By Assumption 3 the expected displacement at T is therefore at least 2δ for every L, and by Assumption 4 it exceeds δ with probability at least $1 - \eta$ once $N_{\text{sleeper}} \geq n_0$. Meanwhile $p_L \to 0$ as $L \to \infty$, so by Assumption 5 the shift within any window of length w vanishes, and by the location and scale invariance behind Lemma 1 the alarm probability approaches $\alpha^{(t)}$. Reachable targets are capped at $\delta_{\max}$, and the confidence $1 - \eta$ comes from the number of sleeper agents rather than from the length of the attack.

Since the iso-mirror is derived from the TDKPS embeddings of *all* agents, displacing it implies that non-adversarial agents' perspectives shift correspondingly.

Corollary 1 (Perspective contamination)**.** *Under the conditions of Theorem 1, for any $\delta \in (0, \delta_{\max})$ and $\eta \in (0, 1)$ with $N_{\text{sleeper}} \geq n_0(\delta, \eta)$, there exists a defection schedule such that, with probability at least $1 - \eta$,*

$$\left\| \frac{1}{N_{\text{honest}}} \sum_{n \notin \mathcal{F}_{\text{sleeper}}} \psi_{n,\pi}^{(T)} - \frac{1}{N_{\text{honest}}} \sum_{n \notin \mathcal{F}_{\text{sleeper}}} \psi_{n,0}^{(T)} \right\|_2 \geq h(\delta),$$

where $\psi_{n,\pi}^{(T)}$ and $\psi_{n,0}^{(T)}$ denote the perspective of agent n under schedule π and under no defection, $h : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ is strictly increasing, and $N_{\text{honest}} = N - N_{\text{sleeper}}$.

Corollary 1 is an existence result: for any target displacement, some defection schedule displaces the perspectives of the non-adversarial agents. It does not imply the converse – an observed displacement of the iso-mirror may also arise from non-adversarial agents adapting to environmental change. Since the perspectives are functions of agent responses only, a displacement driven by a change in the dispersion or shape of the response distributions, rather than their location, is still a change in agent behavior.

Theorem 1 and Corollary 1 suggest a vulnerability of adaptive control: the mechanism enabling adaptation to legitimate change creates an attack surface for adversarial drift. We conclude by formalizing a fundamental tradeoff for adaptive control in multi-agent systems:

Theorem 2 (Learning-security tradeoff)**.** *Consider a multi-agent monitoring system with the following two desiderata:*

1. ***Adaptivity****: The procedure does not raise false alarms at an increased rate when agents update in response to genuine environmental change.*
2. ***Security****: The procedure detects adversarial drift.*

Adaptive control satisfies (1) but fails (2) by Theorem 1. Static control satisfies (2) but fails (1): any legitimate shift exceeding $k\sigma_0$ triggers a false alarm with probability approaching 1.

We conjecture that, under appropriate regularity conditions, no monitoring procedure can simultaneously satisfy both desiderata against all adversaries.

6 Discussion

We introduced adaptive control charts for monitoring multi-agent systems via the iso-mirror and demonstrated a fundamental tradeoff: adaptive control is necessary for monitoring systems whose agents learn from a dynamic environment, yet adaptive control is vulnerable to adversaries that defect sufficiently slowly (Theorem 2). This tradeoff is not an artifact of our particular pipeline – it arises from the location-invariance of any monitoring statistic that re-estimates its baseline from recent data. Any system that adapts to legitimate change will, by the same mechanism, adapt to adversarial change that arrives slowly enough.

Implications for deployed systems. Our results suggest that monitoring multi-agent systems requires more than a single control chart. Detectors sensitive to cumulative rather than instantaneous deviation, such as CUSUM [19], do not escape the tradeoff when their reference is estimated from recent data (§5), but they require a different defection schedule to evade. A schedule tuned to slip past an adaptive Shewhart chart need not also slip past a windowed CUSUM, so running the two together on the iso-mirror trajectory may narrow the set of schedules that evade both.

More broadly, we conjecture that the learning-security tradeoff reflects an adversarial game: for any fixed monitoring procedure, there exists an adversarial strategy that evades it, and conversely, for any fixed adversarial strategy, there exists a monitoring procedure that detects it. Theorem 1 establishes one direction of this conjecture for adaptive control. The reverse direction (that any specific slow-defection schedule can be caught by a suitably designed detector) is not yet formalized.

Online monitoring. Our TDKPS and iso-mirror are fit jointly on all timepoints. Online use requires fixing the coordinates as new timepoints arrive, either by out-of-sample embedding against a reference period [3, 31] or, for large systems, by refitting on bounded windows aligned via Procrustes to a fixed reference. Both rely on a reference period, which exposes a new attack surface: an adversary active during it shapes the coordinates against which later behavior is judged.

Limitations and future work. Several aspects of our setup are simplified relative to real deployments. *(i) Complete communication graphs.* Our simulations use a complete graph with uniform edge weights. Real systems have heterogeneous, sparse, and evolving topologies, whose effect on adversarial propagation speed and detection has not been explored. *(ii) Simple adversary.* We do not address coordinated adversaries or adversaries that adapt to the monitoring system. *(iii) Simplified agents.* Our agents use a simple retrieval-augmented architecture with a fixed prompt template and a single base model. Real-world agents exhibit more complex cognitive architectures – including planning, tool use, self-reflection, and heterogeneous base models – which may affect information propagation dynamics and iso-mirror sensitivity. *(iv) Black-box setting.* Our methodology observes only query-response pairs. Richer observation models (internal activations, reasoning traces, tool calls, or memory writes) may detect adversarial drift earlier. We expect that monitoring systems with such access will exhibit the learning-security tradeoff, though this is an open question. More restrictive access such as view-only (as in Moltbook [23] or other proposed agent ecnomomies [29] may greatly restrict detection possibilities. *(v) Agent-level control.* Our control charts monitor the system-level iso-mirror trajectory and do not localize which agent is responsible for a detected shift. Agent-level monitoring (e.g., applying adaptive control charts to individual trajectories $\psi_n^{(t)}$) may identify adversarial agents more directly, but will face the same learning-security tradeoff compounded by a multiple testing burden across agents.

On the theoretical side, Assumption 1 (i.i.d. stationarity) is not observed in practice or simulation (see Figure 2) and is stronger than necessary. The results extend to mixing processes at the cost of more involved concentration arguments. Further, Assumptions 3–5 are stated as conditions rather than derived from observed dynamics. Characterizing them for specific dissimilarities via perturbation theory remains open, as does characterizing the contamination function h in Corollary 1. Finally, developing defenses theoretically robust to slow adversarial drift is a natural next step.

Acknowledgments

We gratefully acknowledge funding from Defense Advanced Research Projects Agency (DARPA) Artificial Intelligence Quantified (AIQ) award number HR00112520026. We also thank Daniel Kuntz for useful discussions regarding the theory of optimal defection and prompt injection construction.

References

[1] A. Acharyya, M. W. Trosset, C. E. Priebe, and H. S. Helm. Consistent estimation of generative model representations in the data kernel perspective space, 2025. URL https://arxiv.org/abs/2409.17308.

[2] A. Athreya, Z. Lubberts, Y. Park, and C. Priebe. Euclidean mirrors and dynamics in network time series. *Journal of the American Statistical Association*, 120(550):1025–1036, 2025.

[3] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet. Out-of-sample extensions for LLE, Isomap, MDS, Eigenmaps, and spectral clustering. In *Advances in Neural Information Processing Systems 16*, 2003.

[4] E. Bridgeford and H. Helm. Detecting perspective shifts in multi-agent systems, 2025. URL https://arxiv.org/abs/2512.05013.

[5] G. Chen, J. Arroyo, A. Athreya, J. Cape, J. T. Vogelstein, Y. Park, C. White, J. Larson, W. Yang, and C. E. Priebe. Multiple network embedding for anomaly detection in time series of graphs. *Computational Statistics & Data Analysis*, 203:108070, 2024. doi: 10.1016/j.csda.2024.108070.

[6] T. Chen, Z. Lubberts, A. Athreya, Y. Park, and C. E. Priebe. Euclidean mirrors and first-order changepoints in network time series. *arXiv preprint arXiv:2405.11111*, 2024.

[7] W. Choi and J. Kim. Unsupervised learning approach for anomaly detection in industrial control systems. *Applied System Innovation*, 7(1):18, 2024.

[8] Y.-S. Chuang, A. Goyal, N. Harlalka, S. Suresh, R. Hawkins, S. Yang, D. Shah, J. Hu, and T. Rogers. Simulating opinion dynamics with networks of LLM-based agents. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3326–3346, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.211.

[9] B. Duderstadt, H. S. Helm, and C. E. Priebe. Comparing foundation models using data kernels, 2024. URL https://arxiv.org/abs/2305.05126.

[10] L. Hammond, A. Chan, J. Clifton, J. Hoelscher-Obermaier, A. Khan, E. McLean, C. Smith, W. Barfuss, J. Foerster, T. Gavenčiak, et al. Multi-agent risks from advanced AI. Technical Report 1, Cooperative AI Foundation, 2025.

[11] H. Helm, B. Duderstadt, and C. E. Priebe. Tracking perspectives of interacting language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2024.

[12] H. Helm, A. Acharyya, Y. Park, B. Duderstadt, and C. E. Priebe. Statistical inference on black-box generative models in the data kernel perspective space. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 3955–3970. Association for Computational Linguistics, 2025.

[13] E. Hubinger, C. Denison, J. Mu, M. Lambert, M. Tong, M. MacDiarmid, T. Lanham, D. M. Ziegler, T. Maxwell, N. Cheng, et al. Sleeper agents: Training deceptive llms that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024.

[14] M. Huh, B. Cheung, T. Wang, and P. Isola. The platonic representation hypothesis, 2024. URL https://arxiv.org/abs/2405.07987.

[15] M. Kinniment, L. Sato, H. Du, B. Goodrich, M. Winsor, B. Shlegeris, and A. Gabriel. Evaluating language-model agents on realistic autonomous tasks. *arXiv preprint arXiv:2312.11671*, 2023.

[16] H. McGuinness, T. Wang, C. E. Priebe, and H. Helm. Investigating social alignment via mirroring in a system of interacting language models. 2025.

[17] Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar. Nomic embed: Training a reproducible long context text embedder, 2025. URL https://arxiv.org/abs/2402.01613.

[18] OpenAI. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[19] E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954. doi: 10.2307/2333009.

[20] J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*, pages 1–22, San Francisco, CA, USA, 2023. ACM. doi: 10.1145/3586183.3606763.

[21] T. Putterman, D. Lim, Y. Gelberg, S. Jegelka, and H. Maron. Learning on loras: Gl-equivariant processing of low-rank weight spaces for large finetuned models, 2024. URL `https://arxiv.org/abs/2410.04207`.

[22] P. Qiu. *Introduction to Statistical Process Control*. Chapman and Hall/CRC, 2013.

[23] schlichtm. Moltbook: A social network for moltbots, 2025. URL `https://www.moltbook.com/`. Social network platform designed for AI agent interaction without direct human involvement.

[24] S. Schmidl, P. Wenig, and T. Papenbrock. Anomaly detection in time series: A comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022. doi: 10.14778/3538598.3538602.

[25] N. Shapira, C. Wendler, A. Yen, G. Sarti, K. Pal, O. Floody, A. Belfki, A. Loftus, A. R. Jannali, N. Prakash, J. Cui, G. Rogers, J. Brinkmann, C. Rager, A. Zur, M. Ripa, A. Sankaranarayanan, D. Atkinson, R. Gandikota, J. Fiotto-Kaufman, E. Hwang, H. Orgad, P. S. Sahil, N. Taglicht, T. Shabtay, A. Ambus, N. Alon, S. Oron, A. Gordon-Tapiero, Y. Kaplan, V. Shwartz, T. R. Shaham, C. Riedl, R. Mirsky, M. Sap, D. Manheim, T. Ullman, and D. Bau. Agents of chaos, 2026. URL `https://arxiv.org/abs/2602.20021`.

[26] W. A. Shewhart. *Economic Control of Quality of Manufactured Product*. D. Van Nostrand Company, New York, 1931.

[27] P. Steinberger and OpenClaw Contributors. OpenClaw: Your own personal AI assistant, 2025. URL `https://github.com/openclaw/openclaw`. Formerly known as Moltbot/Clawdbot.

[28] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. doi: 10.1126/science.290.5500.2319.

[29] N. Tomašev, M. Franklin, J. Jacobs, S. Krier, and S. Osindero. Distributional AGI safety, 2025.

[30] W. S. Torgerson. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17(4):401–419, 1952. doi: 10.1007/BF02288916.

[31] M. W. Trosset and C. E. Priebe. The out-of-sample problem for classical multidimensional scaling. *Computational Statistics & Data Analysis*, 52(10):4635–4642, 2008.

[32] E. L. Wang, M. S. Kiasari, T. Wang, H. Helm, A. Athreya, C. Priebe, and V. Lyzinski. Gaussian mixture models as a proxy for interacting language models, 2026. URL `https://arxiv.org/abs/2506.00077`.

[33] L. Yee, M. Chui, R. Roberts, and S. Xu. Why agents are the next frontier of generative AI. *McKinsey Quarterly*, July 2024. URL `https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/why-agents-are-the-next-frontier-of-generative-ai`.

A Additional Experimental Details

A.1 Prompt template and examples

Every non-adversarial agent in the swarm queries its language model with the same two-message template (system prompt followed by user prompt). The user prompt is constructed by retrieving the top-k question-answer pairs from the agent's local vector database, formatting each pair as a `Q:/A:` block, joining the blocks with a blank line, and substituting the result into `{retrieved_context}`. Completions are sampled at `temperature = 0.0` with `max_tokens = 500`.

Template.

```
[system]
You answer questions from a database.

[user]
Database results:
{retrieved_context}

Question: {question}

Return the stored answer, or "I don't know" if no entry matches.
```

Example 1: populated prompt ($k = 3$)**.** For the query *"who wrote the declaration of independence"* the agent's retrieval yields three pairs and the user prompt becomes:

```
Database results:
Q: who wrote the declaration of independence
A: Thomas Jefferson

Q: who signed the declaration of independence
A: 56 delegates to the Second Continental Congress

Q: when was the declaration of independence adopted
A: July 4, 1776

Question: who wrote the declaration of independence

Return the stored answer, or "I don't know" if no entry matches.
```

A grounded response would read *"Thomas Jefferson"*, drawing on the first retrieved pair.

Example 2: empty retrieval. When no entries are retrieved, the literal string `(No relevant information found)` is substituted in place of the formatted context, and the model is expected to return `I don't know`:

```
Database results:
(No relevant information found)

Question: what is the population of proxima centauri b

Return the stored answer, or "I don't know" if no entry matches.
```

A.2 No-LLM Proxy for Large-Scale Replications

In order to study the probability of detection under various adversarial conditions, we simulate 100 different replications of three different adversarial profiles: no adversary, a fast attack, and a slow attack. Due to the cost prohibitiveness of running several hundred simulations with an LLM in the loop, we design an experimental setup that captures the information propagation dynamics of the LLMs without needing to call their API explicitly. In particular, we replace the LLM call in our existing simulations with the proxy function detailed in Algorithm 1.

Informally, Algorithm 1 assumes that non-adversarial agents will reproduce the correct answer to a question when they have it in their context, and respond with "I don't know" when they do not have the correct answer in their context. Further, Algorithm 1 assumes that non-adversarial agents will reproduce an adversarial answer retrieved for the query with probability $p_{\text{adv-prop}}$, and otherwise return the highest-scoring answer stored for the query, which may itself be adversarial. In practice, we observed non-adversarial LLMs propagate our chosen canary $\sim 80\%$ of the time, and set $p_{\text{adv-prop}} = 0.8$ as a result. Figure 5 compares system evolution when using an LLM and when using Algorithm 1. The dynamics are qualitatively similar and the accuracy of the agents are nearly identical.



Figure 5: Agent-level dynamics when using LLM agents (blue) and retrieval-only proxy functions (red; Alg. 1). The LLM dynamics and proxy dynamics are qualitatively similar. The trajectories of the mean accuracy (bottom) of the responses are nearly identical.

Algorithm 1 Retrieval-only proxy answer function

Require: Agent A with vector DB $\mathcal{D}$; time t; query q; retrieval cap k; propagation probability $p_{\text{adv-prop}}$; adversarial string $a^\star$.

if A is adversarial **and** $\text{Bernoulli}(p_{\text{defect}}(t))$ **then**
  return $a^\star$ ▷ sigmoid-ramp defection for adversaries
end if
$R \leftarrow \text{TopK}(\mathcal{D}, q, k, t)$
for all $(q', a') \in R$ in score order **do**
  if $a' = a^\star$ **then**
    if $q' = q$ **and** $\text{Bernoulli}(p_{\text{adv-prop}})$ **then**
      return $a^\star$ ▷ infected entry propagates
    end if
    break
  end if
end for
for all $(q', a') \in R$ in score order **do**
  if $q' = q$ **then**
    return a' ▷ highest-scoring stored answer
  end if
end for
return `"I don't know"`

B Theoretical Results

B.1 Standard Control Chart Properties

We present standard properties of the control chart for completeness. Formal definitions of in-control and out-of-control are given in § 5.

Proposition 1 (False alarm rate under normality)**.** *If $F_0 = \mathcal{N}(\mu_0, \sigma_0^2)$ and the system is in-control at time t, then*

$$\alpha^{(t)} = P(|T_w| > k)\,, \qquad T_w \overset{d}{=} \sqrt{\frac{w+1}{w}} \cdot t_{w-1},$$

where t_{w-1} denotes a random variable with the Student t-distribution on $w-1$ degrees of freedom. In particular, $\alpha^{(t)}$ depends on w and k but not on t, μ_0, or σ_0.

Proof. Under in-control conditions, $S^{(t)}, S^{(t-1)}, \ldots, S^{(t-w)}$ are i.i.d. $\mathcal{N}(\mu_0, \sigma_0^2)$. The standardized statistic $(S^{(t)} - \hat{\mu}^{(t)})/\hat{\sigma}^{(t)}$ is invariant to location and scale, so we may assume without loss of generality that $\mu_0 = 0$ and $\sigma_0 = 1$.

Write $Z_0 = S^{(t)}$ and $Z_i = S^{(t-i)}$ for $i = 1, \ldots, w$. Then $\hat{\mu}^{(t)} = \bar{Z}$ and $\hat{\sigma}^{(t)} = \hat{\sigma}_Z$, where $\bar{Z} = w^{-1}\sum_{i=1}^{w} Z_i$ and $\hat{\sigma}_Z^2 = (w-1)^{-1}\sum_{i=1}^{w}(Z_i - \bar{Z})^2$. Since Z_0 is independent of $(\bar{Z}, \hat{\sigma}_Z)$ – the latter being functions of $Z_1, \ldots, Z_w$ only – we have

$$Z_0 - \bar{Z} \sim \mathcal{N}\big(0, \tfrac{w+1}{w}\big)\,,$$

independently of $\hat{\sigma}_Z^2 \sim \chi_{w-1}^2/(w-1)$. By the definition of the t-distribution,

$$\frac{(Z_0 - \bar{Z})/\sqrt{(w+1)/w}}{\hat{\sigma}_Z} \sim t_{w-1},$$

so $(Z_0 - \bar{Z})/\hat{\sigma}_Z \sim \sqrt{(w+1)/w} \cdot t_{w-1}$. The alarm probability is therefore

$$\alpha^{(t)} = P\Big(|t_{w-1}| > k\sqrt{\tfrac{w}{w+1}}\Big)\,,$$

which depends only on w and k. □

Remark 1. *For large w, $\alpha^{(t)} \approx 2(1 - \Phi(k))$; the classical choice $k = 3$ gives $\alpha \approx 0.0027$.*

Corollary 2 (Finite detection horizon)**.** *Under the conditions of Lemma 1:*

1. *For $t \in [t^*, t^* + w)$, the window contains a mixture of pre-shift and post-shift observations, and $1 - \beta^{(t)}(\delta) > \alpha^{(t)}$ whenever $|\delta|$ is sufficiently large relative to σ_0.*
2. *For $t \geq t^* + w$, $1 - \beta^{(t)}(\delta) = \alpha^{(t)}$.*

Proof. Part (2) is immediate from Lemma 1.

For part (1), when $t \in [t^*, t^* + w)$, the window $\mathcal{W}^{(t)}$ contains $t - t^*$ post-shift observations and $w - (t - t^*)$ pre-shift observations. Consequently, $\hat{\mu}^{(t)} = \mu_0 + \frac{t-t^*}{w}\delta + O_p(w^{-1/2})$ and $\hat{\sigma}^{(t)} = \sigma_0 + O_p(w^{-1/2})$. The current observation $S^{(t)}$ has mean $\mu_0 + \delta$, so the effective signal-to-noise ratio is

$$\frac{\mu_0 + \delta - \hat{\mu}^{(t)}}{\hat{\sigma}^{(t)}} \approx \frac{\left(1 - \frac{t-t^*}{w}\right)\delta}{\sigma_0},$$

which is strictly positive for $t \in [t^*, t^* + w)$ and vanishes at $t = t^* + w$. Power therefore exceeds $\alpha^{(t)}$ throughout $[t^*, t^* + w)$ for $|\delta|$ sufficiently large. □

Remark 2. *Lemma 1 relies on the assumption that F_δ is a pure location shift of F_0. If the shift also changes the variance or higher moments, the alarm rate for $t \geq t^* + w$ may differ from $\alpha^{(t)}$. We exploit the pure-location-shift case in the adversarial analysis, where a strategic adversary can match all moments except the mean.*

B.2 Proof of Lemma 1

Proof. For $t \geq t^* + w$, the window $\mathcal{W}^{(t)} = \{t-w, \ldots, t-1\}$ lies entirely in $\{t^*, t^*+1, \ldots\}$, so $S^{(\tau)} \overset{\text{i.i.d.}}{\sim} F_\delta$ for all $\tau \in \mathcal{W}^{(t)}$ and $S^{(t)} \sim F_\delta$ independently.

Since $F_\delta(\cdot) = F_0(\cdot - \delta)$, the joint distribution of $(S^{(t)}, \{S^{(\tau)}\}_{\tau \in \mathcal{W}^{(t)}})$ under the shifted process is identical (up to a global location shift of δ) to the joint distribution under in-control conditions. The alarm criterion $S^{(t)} \notin [\hat{\mu}^{(t)} - k\hat{\sigma}^{(t)},\ \hat{\mu}^{(t)} + k\hat{\sigma}^{(t)}]$ is location-invariant: the statistic $(S^{(t)} - \hat{\mu}^{(t)})/\hat{\sigma}^{(t)}$ has the same distribution whether the common mean is μ_0 or $\mu_0 + \delta$. Therefore,

$$P\Big(A^{(t)} = 1 \mid \text{all window and current observations from } F_\delta\Big) = P\Big(A^{(t)} = 1 \mid \text{in-control}\Big) = \alpha^{(t)}.$$

□

B.3 Proof of Theorem 1

Fix $\delta \in (0, \delta_{\max})$ and $\eta \in (0,1)$, and suppose $N_{\text{sleeper}} \geq n_0(\delta, \eta)$. For $L \geq 2\delta/(C_1 w)$, let every sleeper agent follow

$$\pi_L(t) = p_L \mathbb{1}\{t^* \leq t < t^* + Lw\}, \qquad p_L := \frac{2\delta}{C_1 L w} \in (0,1].$$

This schedule is constant on an interval of length $\tau = Lw$ with $\tau p_L = 2\delta/C_1 < B$, so Assumptions 3–5 apply to it with $p = p_L$. Let $T = t^* + Lw$.

Step 1: Displacement. By Assumption 3, $\mathbb{E}[\Delta_\pi^{(T)}] \geq C_1 \tau p_L = 2\delta$. Hence $\{\Delta_\pi^{(T)} < \delta\} \subseteq \{\Delta_\pi^{(T)} - \mathbb{E}[\Delta_\pi^{(T)}] < -\delta\}$, and Assumption 4 gives

$$P\big(\Delta_\pi^{(T)} < \delta\big) \leq 2\exp\Big(-\frac{N_{\text{sleeper}}\,\delta^2}{2\sigma_a^2}\Big) \leq \eta,$$

the last step because $N_{\text{sleeper}} \geq n_0(\delta, \eta)$. This holds for every admissible L, which gives condition 1 of Definition 3. No independence across windows is used.

Step 2: Perturbation within a window. Fix $t \in [t^*, T]$. The alarm at time t depends only on the $w+1$ values $S_\pi^{(\tau)}, \tau \in \{t-w, \ldots, t\}$. Write $S_\pi^{(\tau)} = S_0^{(\tau)} + d^{(\tau)}$ with $d = m + \xi$ as in Assumption 5. The standardized statistic $(S^{(t)} - \hat{\mu}^{(t)})/\hat{\sigma}^{(t)}$ is unchanged when the same constant is added to all $w+1$ values, so we may replace $S_\pi^{(\tau)}$ by $S_0^{(\tau)} + r^{(\tau)}$ with $r^{(\tau)} := d^{(\tau)} - d^{(t-w)}$. Since d vanishes before t^*, the following bounds also hold when $t - w < t^*$:

$$\max_{t-w \leq \tau \leq t} |r^{(\tau)}| \leq C_3\, p_L w + M_t, \qquad M_t := \max_{0 \leq j \leq w} \big|\xi^{(t-w+j)} - \xi^{(t-w)}\big|.$$

By the maximal Azuma–Hoeffding inequality for martingales with conditionally sub-Gaussian increments, $P(M_t \geq u) \leq 2\exp\big(-u^2/(2wC_3^2 p_L)\big)$. Note that $wp_L = 2\delta/(C_1 L) \to 0$ as $L \to \infty$. Setting $u_L := (wp_L)^{1/4}$ and $\rho_L := C_3 wp_L + u_L$ gives $\rho_L \to 0$ and

$$P\Big(\max_{t-w \leq \tau \leq t} |r^{(\tau)}| > \rho_L\Big) \leq 2\exp\Big(-\frac{1}{2C_3^2 (wp_L)^{1/2}}\Big) =: \kappa_L \to 0.$$

Step 3: Alarm probability. For a current value x_0 and window $x_1, \ldots, x_w$, let $\bar{x}$ and $\hat{\sigma}(x)$ be the window mean and standard deviation (denominator $w-1$). If $\max_i |r_i| \leq \rho$, then $|(x_0 + r_0) - \overline{(x+r)}| \leq |x_0 - \bar{x}| + 2\rho$, and by Minkowski's inequality $\hat{\sigma}(x+r) \geq \hat{\sigma}(x) - \hat{\sigma}(r) \geq \hat{\sigma}(x) - \rho\sqrt{w/(w-1)}$. Hence, on $\{\max|r| \leq \rho_L\}$, an alarm under π at time t implies

$$\big|S_0^{(t)} - \hat{\mu}_0^{(t)}\big| > k\,\hat{\sigma}_0^{(t)} - c_w \rho_L, \qquad c_w := 2 + k\sqrt{w/(w-1)},$$

where $\hat{\mu}_0^{(t)}$ and $\hat{\sigma}_0^{(t)}$ are computed from the counterfactual window. With $Z_0^{(t)} := (S_0^{(t)} - \hat{\mu}_0^{(t)})/\hat{\sigma}_0^{(t)}$, for any $s > 0$,

$$P\big(A^{(t)} = 1\big) \leq P\big(|Z_0^{(t)}| > k - c_w\rho_L/s\big) + P\big(\hat{\sigma}_0^{(t)} < s\big) + \kappa_L.$$

By Assumption 1 the counterfactual values are i.i.d. F_0, so the joint law of $(Z_0^{(t)}, \hat{\sigma}_0^{(t)})$ does not depend on t. Let $G(x) := P(|Z_0^{(t)}| > k - x)$, so that $G(0) = \alpha^{(t)}$. Because $S_0^{(t)}$ is independent of its window and F_0 is continuous, $Z_0^{(t)}$ has no atom at $\pm k$, so $G(x) \downarrow \alpha^{(t)}$ as $x \downarrow 0$. Also $\hat{\sigma}_0^{(t)} > 0$ almost surely, so $P(\hat{\sigma}_0^{(t)} < s) \to 0$ as $s \downarrow 0$. Define

$$\epsilon_L := \inf_{s>0} \Big\{ G(c_w \rho_L / s) - \alpha^{(t)} + P\big(\hat{\sigma}_0^{(t)} < s\big) \Big\} + \kappa_L.$$

Given $\epsilon' > 0$, choose s with $P(\hat{\sigma}_0^{(t)} < s) < \epsilon'/3$, then L large enough that $G(c_w \rho_L / s) - \alpha^{(t)} < \epsilon'/3$ and $\kappa_L < \epsilon'/3$. Hence $\epsilon_L \to 0$ and $P(A^{(t)} = 1) \le \alpha^{(t)} + \epsilon_L$ for all $t \in [t^*, T]$, uniformly in t. This is condition 2 of Definition 3.

Steps 1 and 3 together give (δ, η)-undetected defection at $T = t^* + Lw$, with $\epsilon_L \to 0$ as $L \to \infty$. □

Remark 3 (Speed-stealth tradeoff)**.** *The attack lasts Lw steps. The displacement condition no longer constrains L; it constrains N_{sleeper}. The stealth condition requires the within-window perturbation, which for large L is dominated by the martingale term of order $C_3\sqrt{wp_L} = C_3\sqrt{2\delta/(C_1 L)}$, to be small relative to σ_0. If γ maps a standardized perturbation to the resulting excess alarm probability, it suffices heuristically that $C_3\sqrt{2\delta/(C_1 L)} \lesssim \sigma_0 \gamma^{-1}(\epsilon)$, giving $L^* = \Theta\big(C_3^2 \delta / (C_1 \sigma_0^2 \gamma^{-1}(\epsilon)^2)\big)$ and an attack time of $\Theta\big(w C_3^2 \delta / (C_1 \sigma_0^2 \gamma^{-1}(\epsilon)^2)\big)$. Under Gaussian assumptions, $\gamma(x) \approx 2\big(\Phi(k + x\sqrt{w/(w+1)}) - \Phi(k)\big)$, so $\gamma^{-1}(\epsilon) \approx \epsilon/(2\phi(k))$ for small ϵ and the attack time scales as $\Theta\big(w C_3^2 \delta\, \phi(k)^2 / (C_1 \sigma_0^2 \epsilon^2)\big)$, where ϕ is the standard normal density.*

B.4 Proof of Corollary 1

The iso-mirror distance $\|\phi_\pi^{(T)} - \phi_0^{(T)}\|_2 \ge \delta$ (which holds with probability at least $1-\eta$ by Theorem 1) implies that the distribution of TDKPS embeddings at time T under π differs substantially from that under no defection. Since the iso-mirror is constructed from the dissimilarity $\delta(\cdot,\cdot)$ between agent embedding distributions at different timepoints, a shift of magnitude δ in the iso-mirror implies a corresponding change in the agent embedding distribution.

The sleeper agents form a minority ($N_{\text{sleeper}} < N$). Decompose the centroid of all agents as

$$\bar{\psi}^{(t)} = \frac{N_{\text{honest}}}{N} \bar{\psi}^{(t)}_{\text{honest}} + \frac{N_{\text{sleeper}}}{N} \bar{\psi}^{(t)}_{\text{sleeper}},$$

where $\bar{\psi}^{(t)}_{\text{honest}} = N_{\text{honest}}^{-1} \sum_{n \notin \mathcal{F}_{\text{sleeper}}} \psi_n^{(t)}$. If the overall embedding distribution shifts (as implied by the iso-mirror displacement) and the sleeper agents contribute at most N_{sleeper}/N of the total, then the honest agents must account for the remaining shift:

$$\Big\| \bar{\psi}^{(T)}_{\text{honest},\pi} - \bar{\psi}^{(T)}_{\text{honest},0} \Big\|_2 \ge h(\delta),$$

where h depends on the dissimilarity used in the iso-mirror construction, the ratio N_{sleeper}/N, and the geometry of the embedding space. For instance, if the dissimilarity is the energy distance between embedding distributions, $h(\delta)$ is bounded below by a function that is strictly increasing in δ and decreasing in N_{sleeper}/N. □

B.5 Proof of Theorem 2

Adaptive control fails security. This is Theorem 1: for any $\delta \in (0, \delta_{\max})$ and $\eta \in (0,1)$, if $N_{\text{sleeper}} \ge n_0(\delta, \eta)$ there exists a defection schedule achieving (δ, η)-undetected defection in finite time.

Static control fails adaptivity. Under static control, control limits are $\text{LCL} = \hat{\mu}_0 - k\hat{\sigma}_0$ and $\text{UCL} = \hat{\mu}_0 + k\hat{\sigma}_0$, where $\hat{\mu}_0$ and $\hat{\sigma}_0$ are estimated from the burn-in period $\{1, \dots, T_0\}$ and held fixed. If the environment changes so that $\mathbb{E}[S^{(t)}] = \mu_0 + \delta_{\text{env}}$ for $t > T_0$, then by the law of large numbers, as $T_0 \to \infty$, $\hat{\mu}_0 \to \mu_0$ and $\hat{\sigma}_0 \to \sigma_0$. For $|\delta_{\text{env}}| > k\sigma_0$, the probability

$$P(S^{(t)} \notin [\text{LCL}, \text{UCL}]) = P\left(\left| \frac{S^{(t)} - \hat{\mu}_0}{\hat{\sigma}_0} \right| > k \right)$$

approaches 1 as $T_0 \to \infty$, since $S^{(t)}$ has mean $\mu_0 + \delta_{\text{env}}$ while the control limits remain centered at μ_0. This constitutes a false alarm (the change is legitimate), so static control fails adaptivity. □

C Moltbook Scale Simulations

We repeat the detectability analysis of Section 4 at the scale of agent-only networks such as Moltbook [23]. We vary the number of agents $N \in \{10^2, 10^3, 10^4, 10^5\}$, the mean degree of the contact graph, the number of sleeper agents $N_{\text{sleeper}} \in \{1, 10, 100\}$, and the size of the query pool $M \in \{50, 500\}$, which we refer to as information diversity.

Setup. We use a vectorized re-implementation of the proxy in Algorithm 1 ($p_{\text{adv-prop}} = 0.8$), checked against the original simulator. Agents interact over an Erdős–Rényi contact graph with mean degree in $\{9, 99, 999, 9{,}999\}$, or over the complete graph (mean degree $N - 1$). The query pool keeps the composition of the main experiments: 20% environmental queries and 80% static queries, with each static query seeded to exactly one agent. Agents use $k = 3$ retrieval, $K = 5$ questions per interaction and $\rho = 0.05$. Environmental answers change with probability $p_{\text{env}} = 0.04$ per iteration, and each run lasts 2,000 iterations. Every sleeper agent follows the slow sigmoid schedule with $t^* = 400$, $s = 1{,}600$, $p_{\max} = 0.75$ and $\kappa = 1$.

For $N > 100$, TDKPS is computed on a fixed panel of 100 agents that includes a sleeper agent; for $N = 100$ it uses all agents. The panel is queried every 10 iterations on all environmental queries and an equal number of sampled static queries. The iso-mirror is obtained by one-dimensional Isomap, and monitored with an adaptive control chart with $w = 200$ and $k = 3$, evaluated at every snapshot from t^* on (160 evaluations per run).

Metrics. Every configuration is run 30 times with sleeper agents and 30 times without, on matched seeds. *Contamination* ("infected fraction of victims" in Figure 6) is the fraction of non-adversarial agents whose knowledge base holds the canary at the end of the run; it is the empirical counterpart of Corollary 1. The *alarm ratio* is the fraction of chart evaluations out of band in the runs with sleepers, divided by the same fraction in the paired runs without them, pooled over the 30 replications (4,800 evaluations per arm) with add-one smoothing; it is the empirical counterpart of condition 2 of Definition 3.

Results. Figure 6 shows 100 sleeper agents at $M = 500$. Contamination falls as the system grows and rises with connectivity. Five takeovers (contamination above one half) leave the monitor no more active than in a clean network (alarm ratio at most 1.0 after rounding; outlined). In particular, 100 sleeper agents in a 100,000-agent system with mean degree 999, one tenth of one percent of the population, contaminate 57% of the non-adversarial agents with an alarm ratio of 0.65. Every other takeover has an alarm ratio above 1, but these ratios (the 10^3 column, and the 10^4 column at mean degrees 99 and 999) rest on very few alarms, between 1 and 4 with sleepers against 0 or 1 without, and none is significant under a one-sided binomial test at level 0.05. The legend's "detectable" label for these cells should be read in that light.

The figure is one of three configurations we ran. At the information diversity of the main experiments ($M = 50$), a single sleeper agent contaminates every non-adversarial agent at every network size and mean degree tested. There, for $N \geq 10^3$ the attack does raise the alarm rate measurably (ratios 1.4 to 1.8, each significant), so complete takeover coexists with a modest increase in alarms. At $M = 500$ with only 10 sleepers (Table 1), contamination is diluted as the system grows: at mean degree 9 it falls from 1.00 at $N = 10^2$ to 0.01 at $N = 10^5$, while down any column it recovers with connectivity. Raising the cohort to 100 (Figure 6) restores it. With 100 sleepers at $N = 10\,N'$, contamination matches that of 10 sleepers at N' to within 0.01 in every cell where both exist, so contamination tracks the adversarial fraction and the topology rather than the system size.

	N (agents)			
mean degree	10^2	10^3	10^4	10^5
9	1.00	0.62	0.10	0.01
99	1.00	0.95	0.26	0.03
999	–	1.00	0.58	0.08
9,999	–	–	0.70	0.11
99,999	–	–	–	0.11

Table 1: Contamination with 10 sleeper agents at $M = 500$, otherwise as in Figure 6; 30 replications per cell. Dashes mark mean degrees above $N - 1$; the top entry of each column is the complete graph.

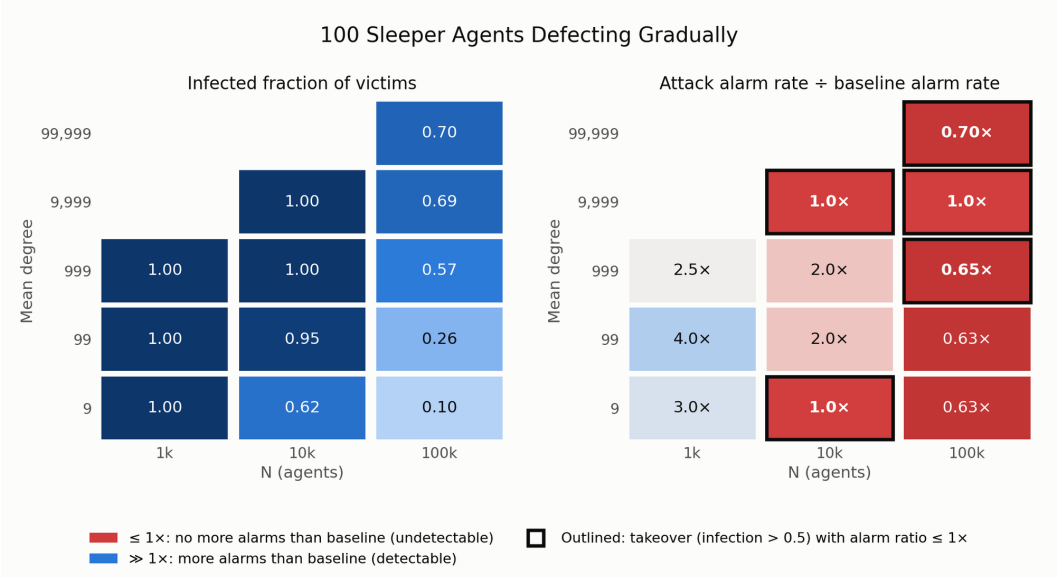

Figure 6: Gradual defection by 100 sleeper agents across network size N (columns) and mean degree (rows; the top cell in each column is the complete graph), with $M = 500$ queries, $t^* = 400$, $s = 1{,}600$ and $p_{\max} = 0.75$. TDKPS is computed on a fixed panel of 100 agents that includes a sleeper agent. **Left**: Fraction of non-adversarial agents holding the canary at $T = 2{,}000$. **Right**: Alarm rate of the adaptive control chart ($w = 200$, $k = 3$) after t^*, relative to matched runs without sleeper agents; 30 replications per cell. Red: ratio at most 1. Outlined: takeovers (contamination above 0.5) with ratio at most 1. The remaining takeovers have ratios above 1 that rest on at most 4 alarms and are not significant (one-sided binomial test, level 0.05). Gradual defection can take over large systems without raising the alarm rate.