JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 11 Issue 02
2026
Research Article
https://doi.org/10.56979/1102/2026/1528

Adaptive Randomized Asynchronous Sampling and Distributed Detection for Resilient Consensus of Heterogeneous Multiagent Systems Under Schedule-Aware Zero-Dynamics Attacks

Muhammad Zunnurain Hussain[1*], Muhammad Zulkifl Hasan[2], and Hafiz Muhammad Usman[3]

[1]Department of Computer Science, Bahria University Lahore, Pakistan.
[2]Faculty of Information Technology, University of Central Punjab, Lahore Pakistan.
[3]Alhamara University DHA Campus, Pakistan.
[*]Corresponding Author: Muhammad Zunnurain Hussain. Email: Zunnurain.bulc@bahria.edu.pk

Received: May 22, 2026 **Accepted:** August 31, 2026

Abstract: Zero-dynamics attacks (ZDAs) inject signals that keep the monitored outputs at their nominal values while the internal states diverge. In sampled-data multiagent systems (MASs) the attack uses unstable invariant zeros of the discretized model. Liu et al. [1] showed that asynchronous sampling with integer-ratio periods removes the destructive sampling zeros of homogeneous second-order MASs. That countermeasure runs on a fixed schedule, and an attacker who reads packet timestamps can recover the schedule, rebuild the sampled model, and attack again. We propose adaptive randomized asynchronous sampling (ARAS) together with a distributed windowed χ^2 detector for heterogeneous MASs with agent-specific measurement coefficients. We show that the invariant zeros of the sampled network split into one zero per agent and give closed-form conditions on the coefficients and the period. When the attacker does not know the current sampling mode at the moment it commits its input, a perfectly stealthy attack survives K consecutive steps with probability at most $(1 - (m - 1)p_min)^K$, and the residual seen by the detector has an explicit lower bound. If the current mode leaks to the attacker, the guarantee disappears, so the secrecy of the draw is part of the design. Mean-square practical consensus under measurement noise follows from a windowed contraction condition that holds for every admissible schedule, independently of the adaptation. In 500 Monte Carlo trials per configuration on a four-quadrotor formation, a schedule-aware attacker succeeded in 99.8% of the runs against deterministic asynchronous sampling and in 5.6% (95% CI 3.9% to 8.0%) against ARAS. ARAS sent 29.7% more control messages than synchronous sampling, against 49.0% for the deterministic scheme. We also report ablations, detector operating curves, and tests with up to 50 agents, four topologies, four noise levels, several attack strengths, and partially informed attackers.

Keywords: Zero-dynamics attack; multiagent systems; asynchronous sampling; moving-target defense; resilient consensus; distributed detection; heterogeneous agents; cyber-physical security

1. Introduction

Sampled-data multiagent systems (MASs) are integral to unmanned aerial vehicle (UAV) formations, robotic swarms, distributed energy networks, and intelligent transportation [1]–[3]. Their reliance on shared communication networks, however, exposes them to malicious cyber attacks. Among these, the zero-dynamics attack (ZDA) is particularly insidious: by aligning the attack signal with the weakly unobservable subspace of the plant, the attacker keeps the output residual at its nominal value even as the internal states diverge [4], [5].

Sampled-data implementations compound this risk. Even minimum-phase continuous-time plants acquire unstable sampling zeros upon discretization with a zero-order hold (ZOH) [6], [7]. Liu et al. [1] recently established a quantitative relationship between these sampling zeros and two design parameters, the sampling period T and the measurement matrix C, and proposed a deterministic asynchronous sampling countermeasure that eliminates all destructive unstable zeros for homogeneous second-order MASs under the assumption of a connected undirected graph and noise-free measurements.

Heterogeneous measurements change the picture. For a double integrator observed through $y_i = c_{1i}x_i + c_{2i}v_i$ with $c_{1i}c_{2i} < 0$, the continuous-time model already has a zero at $s = -c_{1i}/c_{2i} > 0$, and every sampling period maps it to an unstable discrete zero (Theorem 1). Changing the period moves this zero along the real axis but never brings it inside the unit circle, so a fixed schedule, synchronous or asynchronous, leaves such agents exposed.

Motivation. The asynchronous countermeasure of [1] is analytically powerful, but uses a fixed, publicly known sampling schedule. A schedule-aware attacker who observes packet timestamps can identify the deterministic asynchronous ratios, reconstruct the sampled model, and mount a ZDA against that model, achieving the same damage as against the synchronous system (Proposition 1). A fixed schedule gives the attacker a stationary target. This paper randomizes the schedule and adapts its distribution online.

Novelty with respect to [1]. The work in [1] treats homogeneous agents that share one output matrix, so all agents share one sampling zero, and it removes the destructive zeros by choosing suitable periods. That route is closed for agents with $c_{1i}c_{2i} < 0$, whose zero is unstable for every period. ARAS does not try to make the zero stable. It makes the zero that the attacker must match unpredictable at the moment the attacker acts. This requires four ingredients that are absent from [1]: an explicit information pattern for the attacker, a probabilistic stealth bound that depends on that pattern, a detector with a threshold design that accounts for measurement noise, and a consensus analysis for asynchronous random periods. We also show that the information pattern is necessary: if the attacker learns the current period before it acts, randomization gives no protection (Proposition 3).

Contributions. The paper makes five contributions.

1) Network zero decoupling and heterogeneous characterization (Lemma 1, Theorem 1). Using feedback invariance of invariant zeros, we show that the zeros of the closed-loop sampled network are the union of single-agent zeros, independent of the graph and the gains. We give their closed form, a corrected sign table, and their relation to the continuous-time zero.

2) Stealth and residual bounds under randomized schedules (Theorems 2 and 4, Proposition 3). For any attacker that does not know the current mode, a perfectly stealthy attack lasts K steps with probability at most $(1 - (m - 1)\, p_{min})^K$, and each step produces a residual larger than an explicit margin with probability at least $(m - 1)\, p_{min}$. Both bounds fail if the current mode leaks.

3) ARAS with distributed detection (Algorithm 1): a sampling law with an explicit distribution, floor, and adaptation rule, and a local χ^2 detector whose threshold is set from a per-sample false-alarm probability and calibrated to an alarm rate per agent-hour.

4) Mean-square practical consensus (Theorem 3) for asynchronous, heterogeneous, randomly timed sampling with measurement noise, under a windowed contraction condition that holds for every admissible schedule and therefore for any adaptation of the probabilities.

5) An evaluation with 500 Monte Carlo trials per configuration and 95% confidence intervals, six strategies compared under identical attack, noise, and network conditions (including dual-rate sampling [8] and watermarking [15]), ablations, operating curves of the detector, tests with up to 50 agents, four topologies, damped agents, and attackers with partial or delayed knowledge. The simulation code is provided as supplementary material.

Section II reviews related work. Section III formulates the system and threat models. Section IV extends the vulnerability analysis. Section V presents the ARAS mechanism. Section VI contains the theoretical guarantees, the trade-off analysis, and the cost of the scheme. Section VII presents simulations. Section VIII discusses limitations and stronger attackers, and Section IX concludes.

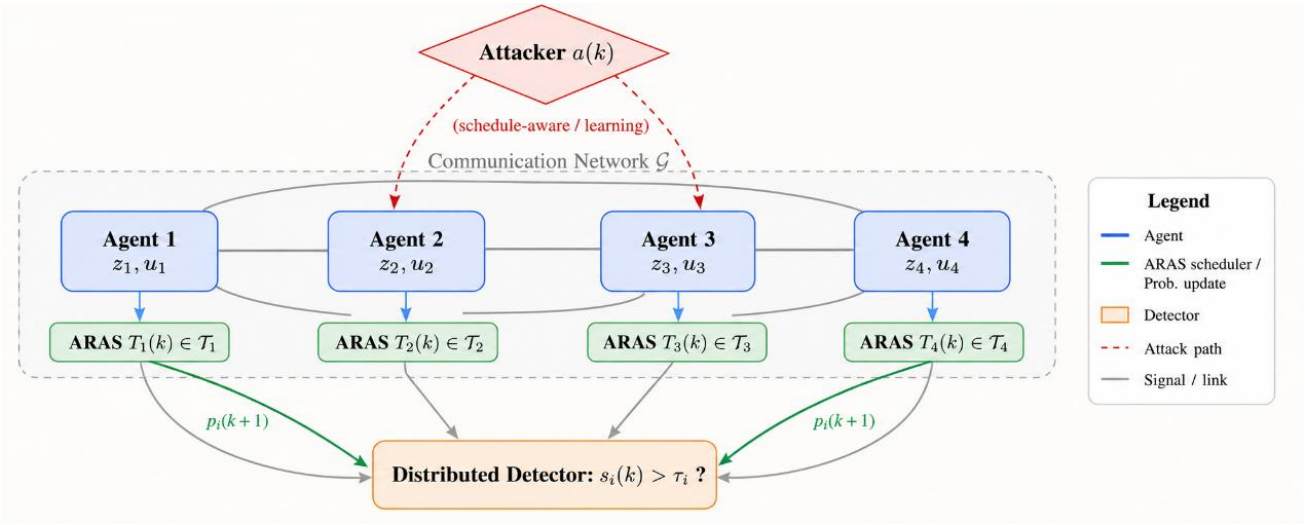


Figure 1. ARAS architecture for four heterogeneous agents connected through graph $\mathcal{G}$. The attacker (classes A3 and A4, Section III-E) alters the command packets of agents 2 and 3.

Each agent runs its own ARAS scheduler, which draws the next sampling period $T_i(k)$ from its mode set $\mathcal{T}_i$, and its own windowed detector, which compares the local statistic $s_i(k)$ with τ_i. The detector of agent i feeds only the probability update $p_i(k+1)$ of agent i. The detector is drawn as one block for compactness; no detector data are exchanged between agents.

2. Related Work

2.1. Zero-Dynamics Attacks

The ZDA threat was formalized by Teixeira et al. [4] who gave necessary and sufficient conditions for vulnerability of general linear systems. Mao et al. [12] proposed new stealthy attack and defense strategies for networked control systems. In [7] the sampled-data case has been discussed and in [11] the ZDA for uncertain systems has been investigated. In [13] the periodic ZDAs of a discrete-time second order MASs have been analyzed. In [26], quantized ZDAs which guarantee that the output error lies within certain bound were developed. We focus on an attacker that adapts to the defense by learning its sampling schedule, a case that has received little attention.

2.2. Sampling-Based Defenses Against ZDA

Multirate sampling was proposed as a countermeasure in [8], [19], relocating unstable sampling zeros to stable positions. In the dual-rate scheme of [8], the output is sampled faster than the ZOH updates the input; the lifted system then has more outputs than inputs and, generically, no invariant zeros. Kim et al. [10] introduced generalized hold functions that move zeros inside the unit disk. Chen et al. [17] extended this to aperiodic sampled-data systems. The direct predecessor of our work is [1], which proved that asynchronous sampling with integer-ratio subperiods eliminates all destructive zeros for homogeneous second-order MASs. All of these methods use deterministic timing. For an attacker that can only alter command packets, deterministic timing is harmless in the dual-rate scheme, because the attacker cannot change its input between two output samples; we include [8] as a baseline for this reason. Deterministic asynchronous sampling [1] keeps a square sampled model whose zero the attacker can compute once the period is known. Moving-target defenses have been studied in broader CPS contexts [14], [21], but not for the multi-agent zero-dynamics attack channel with adaptive attackers.

2.3. Secure Consensus

Resilient consensus using MSR filtering [22], observer-based residual generation [18], [23], and unknown-input observers [24] has been extensively studied. Physical watermarking [15] injects probing signals to detect spoofed outputs, trading off nominal performance for detectability. Against a ZDA built on an accurate model,

watermarking does not help, since the agent removes its own known watermark from the prediction and the attack remains output-nulling; Section VII confirms this. Our approach obtains detectability from timing randomization, without biasing the control signal. Event-triggered secure control [18] reduces communication but does not address the sampling-zero attack channel. Network delay and denial-of-service effects on sampled-data consensus are analyzed in [25], motivating our focus on practical network impairments.

2.4. Data-Driven Intrusion Detection

Machine learning-based intrusion detection has been studied in 5G cellular networks [35], in cloud computing for the detection and mitigation of DDoS attacks [36], and in IoT networks for botnet detection [37]. Convolutional neural networks have also been used to learn intrusion features for cloud security [38]. These detectors work on network traffic and learn attack signatures from labeled data, without a model of the physical plant. A ZDA, in contrast, sends packets that are well formed and correctly timed and only changes the value of the command, and it keeps the physical outputs at their nominal values by design. Traffic-based detectors and the model-based residual test of ARAS therefore look at different evidence, and they can be used together: the former can flag injected or malformed packets, which fall outside Assumption 6, while the latter targets value manipulation of legitimate command packets.

3. System Model and Threat

3.1. Communication Graph and Agent Dynamics

Assumption *1 (Topology):* The communication graph $G = (V, E, A)$ is undirected and connected, with $|V| = N$, adjacency matrix $A = [a_{ij}]$, $a_{ij} = a_{ji} \geq 0$, and Laplacian $L = \Delta - A$, where $\Delta = \mathrm{diag}(\Sigma_j a_{ij})$. Its eigenvalues are $0 = \mu_1 < \mu_2 \leq \ldots \leq \mu_N$. When a message is lost, the receiver uses the last value it received from that neighbor. The N agents are heterogeneous second-order systems

$$\dot{x}_i = v_i, \quad \dot{v}_i = -d_i v_i + b_i u_i, \quad i = 1, \ldots, N, \tag{1}$$

where $x_i, v_i, u_i \in \mathbb{R}^p$. *Assumption 2 (Agent dynamics):* $d_i \geq 0$ and $b_i > 0$. Sections IV to VI use $d_i = 0$ and $b_i = 1$, as in [1]; Remark 3 and Section VII-F treat $d_i > 0$ and $b_i \neq 1$. For clarity the scalar case $p = 1$ is presented; the vector case follows from Kronecker products with I_p [1], [2].

3.2. Sampled-Data Protocol Under Attack

Agent i samples at its own instants t^i_k, with $t^i_{k+1} = t^i_k + T_i(k)$. Under a ZOH, the control protocol under attack is

$$u_i(t) = \alpha \Sigma_{j \in N_i} a_{ij} (\tilde{x_j}(t^i_k) - \tilde{x_i}(t^i_k)) + \beta \Sigma_{j \in N_i} a_{ij} (\tilde{v}_j(t^i_k) - \tilde{v}_i(t^i_k)) + a_i(k) \tag{2}$$

for $t \in [t^i_k, t^i_{k+1})$, $\alpha, \beta > 0$, where $\tilde{x_j}$ and $\tilde{v}_j$ are the noisy states received at t^i_k (Section III-C). When all agents share the period T, stacking $z = [x^\top, v^\top]^\top \in \mathbb{R}^{2N}$ gives the exact discrete-time model

$$z(k+1) = A_d\, z(k) + B_d\, a(k), \tag{3a}$$

$$y(k) = C_d\, z(k), \tag{3b}$$

where $a(k) = [a_1(k), \ldots, a_N(k)]^\top$, and

$$A_d = \begin{bmatrix} I_N & T \cdot I_N \\ 0_N & I_N \end{bmatrix} - \begin{bmatrix} (\alpha/2)T^2 L & (\beta/2)T^2 L \\ \alpha T \cdot L & \beta T \cdot L \end{bmatrix}, \quad B_d = \begin{bmatrix} (T^2/2) \cdot I_N \\ T \cdot I_N \end{bmatrix},$$

$$C_d = [C_1 \;\; C_2], \quad C_\ell = \mathrm{diag}\{c_{\ell 1}, \ldots, c_{\ell N}\}, \quad \ell = 1, 2.$$

The closed-loop matrix is a state feedback applied to N decoupled double integrators:

$$A_d = A_o - B_d K, \quad A_o = \begin{bmatrix} I_N & T \cdot I_N \\ 0_N & I_N \end{bmatrix}, \quad K = [\, \alpha L \;\; \beta L \,]. \tag{3c}$$

The output $y_i(t^i_k) = c_{1i} x_i(t^i_k) + c_{2i} v_i(t^i_k)$ uses agent-specific coefficients (c_{1i}, c_{2i}), extending the homogeneous assumption of [1]. Assumption 3 (Measurement coefficients): $(c_{1i}, c_{2i}) \neq (0, 0)$ and the coefficients are known to agent i. For the agents protected by ARAS, $c_{1i} c_{2i} \neq 0$ and $T c_{1i} + 2 c_{2i} \neq 0$ for every admissible period T. The attacker is assumed to know them (worst case).

3.3. Measurement Noise

Actual measurements are noisy: $\tilde{y}_i(k) = y_i(k) + n_i(k)$, $n_i(k) \sim N(0, \sigma_i^2)$. The states used in (2) carry navigation noise, $\tilde{x_j} = x_j + \nu_j^x$ and $\tilde{v}_j = v_j + \nu_j^v$, with variance σ_v^2. Assumption *4 (Noise):* all noise sequences are i.i.d., zero mean, mutually independent, and independent of the initial states and of the mode draws.

3.4. Admissible Sampling Modes

Assumption *5 (Modes):* Every agent with $c_{1i}c_{2i} < 0$ has a mode set $\mathcal{T}_i = \{T_i^{(1)}, \ldots, T_i^{(m)}\}$ of $m \geq 2$ distinct periods in $(0, \bar{T})$, where $\bar{T}$ is the largest period for which synchronous sampling reaches consensus. For each Laplacian eigenvalue $\mu > 0$, the disagreement mode of (3) has characteristic polynomial $z^2 - (2 - \beta\mu T - \alpha\mu T^2/2)z + 1 - \beta\mu T + \alpha\mu T^2/2$, and the Jury test gives stability if and only if $T < 2\beta/\alpha$ and $T < 2/(\beta\mu)$. When $\beta^2\mu_N < 4\alpha$, this yields $\bar{T} = \min\{2\beta/\alpha, 2/(\beta\mu_N)\}$, which equals 0.625 s in Section VII. Agents with $c_{1i}c_{2i} > 0$ sample with the nominal period T_0. The draw $T_i(k)$ uses a private random source, for example a cryptographic generator with a secret seed, and is independent of everything the attacker has observed up to t^i_k. The sampling clock is trusted: the attacker cannot move sampling instants.

3.5. Attacker Models

Assumption *6 (Attack channel):* For each compromised agent $i \in A_c \subseteq V$, the attacker alters the value of the command packet, so the actuator receives $u_i + a_i(k)$ over $[t^i_k, t^i_{k+1})$. The attacker cannot create packets, delay them, or move sampling instants, and it does not alter sensor data.

Information pattern. When it commits $a_i(k)$ at t^i_k, the attacker knows the plant model, the coefficients, the mode set $\mathcal{T}_i$, every past period $T_i(0), \ldots, T_i(k-1)$ (read from packet timestamps), and its own past inputs. From these it can compute the deviation $e_i(k)$ it has caused (Section V-C). We call this information $\mathcal{I}_i(k)$. The current period $T_i(k)$ is revealed only at t^i_{k+1}, when the next packet arrives. We distinguish five classes:

A1 — Exact-model, schedule-unaware: designs the output-nulling input for the nominal period T_0 of the synchronous system.

A2 — Imperfect-model: behaves as A3 but uses coefficients with a relative error ε ($\varepsilon = 5\%$ in Section VII).

A3 — Schedule-aware: knows $\mathcal{I}_i(k)$ and the current distribution $p_i(k)$ exactly. For a deterministic schedule it therefore knows $T_i(k)$. For a random schedule it either bets on the most likely mode (MAP rule) or chooses the input that minimizes the expected squared next residual (hedging rule); Section VII reports the worse of the two for the defender.

A4 — Schedule-learning: as A3, but replaces $p_i(k)$ by the empirical frequencies of the last M observed periods, possibly with a delay of d intervals.

A5 — Clairvoyant: knows $T_i(k)$ before it commits $a_i(k)$, for example after stealing the seed. A partially informed attacker receives $T_i(k)$ in advance with probability q and otherwise acts as A3. A5 is used to show the limits of the scheme (Proposition 3, Section VIII).

Classes A3 to A5 are outside the threat model of [1] and are the focus of this work.

4. Vulnerability Analysis

4.1. Heterogeneous Zero Characterization

Lemma *1 (Extended from [1, Lemma 1]):* For a common period T and diagonal C_1, C_2, λ is an invariant zero of (A_d, B_d, C_d) in (3) if and only if $\lambda = \lambda_0^{(i)}(T)$ for some agent i, where

$$\lambda_0^{(i)}(T) = -(T c_{1i} - 2c_{2i}) / (T c_{1i} + 2c_{2i}), \quad i = 1, \ldots, N. \tag{4}$$

In particular, the zeros do not depend on L, α, or β.

Proof: By (3c), the Rosenbrock matrix satisfies $[\lambda I - A_d, -B_d; C_d, 0] = [\lambda I - A_o, -B_d; C_d, 0]\cdot U$ with $U = [I, 0; -K, I]$, and U is unimodular, so both pencils have the same invariant zeros (feedback invariance). After a permutation that groups (x_i, v_i, a_i), the open-loop pencil is block diagonal with N blocks $R_i(\lambda) = [\lambda - 1, -T, -T^2/2; 0, \lambda - 1, -T; c_{1i}, c_{2i}, 0]$. Expanding along the last row, $\det R_i(\lambda) = c_{1i}(T^2/2)(\lambda + 1) + c_{2i}T(\lambda - 1) = (T/2)[(Tc_{1i} + 2c_{2i})\lambda + Tc_{1i} - 2c_{2i}]$, whose root is (4). The determinant of the full pencil is the product of the blocks, so it vanishes exactly at the agent zeros. ■

Residual dynamics. The quantity that matters for detection is the residual, and its dynamics are simpler than (3). Let $\Phi(T) = [1, T; 0, 1]$ and $\Gamma(T) = [T^2/2; T]$ be the single-agent ZOH matrices and $C_i = [c_{1i}\ \ c_{2i}]$. Agent i knows the command $u_i(k)$ it sent and the period it used, so its predictor (Section V-C) is driven by $u_i(k)$ and the deviation $e_i = z_i - \hat{z}_i$ obeys

$$e_i(k+1) = \Phi(T_i(k))\, e_i(k) + \Gamma(T_i(k))\, a_i(k), \quad r_i(k) = C_i e_i(k) + n_i(k). \tag{5}$$

Equation (5) holds for asynchronous sampling, where (3) is not defined, and it does not involve the neighbors: the coupling enters $u_i(k)$, which the predictor already contains. Hence a single compromised agent suffices to fool its own detector, the attack does not appear in the residuals of the other agents, and the relevant zero is the single-agent zero (4) evaluated at the period actually in use. The attack still damages the formation, because the true state of agent i diverges and the protocol propagates the deviation to its neighbors.

Two identities follow directly from (5). Let $w_i = [c_{2i}, -c_{1i}]^\top$, which spans ker C_i, and $g_i(T) = C_i\Gamma(T) = T(Tc_{1i} + 2c_{2i})/2$. If $e_i(k) = \xi w_i$, then $C_i\Phi(T)w_i = -Tc_{1i}^2$ and, for any input a,

$$C_i e_i(k+1) = g_i(T)\,[\, a - a^\star_i(T, \xi)\,], \quad a^\star_i(T, \xi) = 2c_{1i}^2\xi \,/\, (Tc_{1i} + 2c_{2i}), \tag{6}$$

$$a = a^\star_i(T, \xi) \implies e_i(k+1) = \lambda_0^{(i)}(T)\,\xi\, w_i. \tag{7}$$

Equation (7) is the zero dynamics of agent i: the output-nulling input keeps the deviation in ker C_i and multiplies it by $\lambda_0^{(i)}(T)$ at every step.

Theorem *1* (Heterogeneous zero positions): The stability of $\lambda_0^{(i)}$ from (4) is governed by the product $c_{1i}c_{2i}$ and the sampling period T as summarized in Table I. Specifically: (i) if $c_{1i}c_{2i} > 0$, then $|\lambda_0^{(i)}| < 1$ for all $T > 0$ and no destructive ZDA based on this agent can exist; (ii) if $c_{1i}c_{2i} < 0$, then $|\lambda_0^{(i)}| > 1$ for every $T > 0$ with $Tc_{1i} + 2c_{2i} \neq 0$; (iii) if $c_{1i}c_{2i} \neq 0$, then $d\lambda_0^{(i)}/dT = -4c_{1i}c_{2i}/(Tc_{1i} + 2c_{2i})^2$, so $T \mapsto \lambda_0^{(i)}(T)$ is strictly monotone on each side of $T = -2c_{2i}/c_{1i}$ and injective on the admissible periods; (iv) with $s_i = c_{1i}/c_{2i}$, $\lambda_0^{(i)}(T) = (1 - s_iT/2)/(1 + s_iT/2)$, the (1,1) Padé approximant of $\exp(-s_iT)$, and $-s_i$ is the zero of the continuous-time transfer function $(c_{2i}s + c_{1i})/s^2$. Heterogeneous coefficients therefore give agent-specific zero loci, and the unstable zero of an agent with $c_{1i}c_{2i} < 0$ is the image of its non-minimum-phase continuous-time zero.

Proof: For $T > 0$, $|\lambda_0^{(i)}| < 1$ if and only if $(2c_{2i} - Tc_{1i})^2 < (2c_{2i} + Tc_{1i})^2$, that is, $-8Tc_{1i}c_{2i} < 0$, which gives (i) and (ii). Differentiating (4) gives (iii); on the two sides of the pole the zero lies in $(1, \infty)$ and $(-\infty, -1)$, so injectivity holds on the whole admissible set. Dividing numerator and denominator of (4) by $2c_{2i}$ gives (iv).

Table 1. Sampling Zero Positions for Agent I (EXTENDS [1, TABLE I] to Heterogeneous Case; Rows 4 TO 6 Corrected or Added)

c_{1i}	c_{2i}	Condition	Zero $\lambda_0^{(i)}$
> 0	> 0	any $T > 0$	$\lvert\lambda_0^{(i)}\rvert < 1$, stable
< 0	> 0	$0 < T < -2c_{2i}/c_{1i}$	$\lambda_0^{(i)} > 1$, unstable
< 0	> 0	$T > -2c_{2i}/c_{1i}$	$\lambda_0^{(i)} < -1$, unstable
> 0	< 0	$0 < T < 2\lvert c_{2i}\rvert/c_{1i}$	$\lambda_0^{(i)} > 1$, unstable
> 0	< 0	$T > 2\lvert c_{2i}\rvert/c_{1i}$	$\lambda_0^{(i)} < -1$, unstable
< 0	< 0	any $T > 0$	$\lvert\lambda_0^{(i)}\rvert < 1$, stable
$= 0$	$\neq 0$	any $T > 0$	$\lambda_0^{(i)} = 1$, boundary
$\neq 0$	$= 0$	any $T > 0$	$\lambda_0^{(i)} = -1$, boundary

At $T = 2\lvert c_{2i}\rvert/\lvert c_{1i}\rvert$ with $c_{1i}c_{2i} < 0$ the zero moves to infinity ($g_i(T) = 0$); Assumption 3 excludes this period.

Remark 1 (Consequence for design): For an agent with $c_{1i}c_{2i} < 0$, no fixed period, synchronous or asynchronous, places the zero inside the unit circle. The approach of [1], which selects periods with harmless zeros, is not available for these agents. Boundary agents ($c_{1i} = 0$ or $c_{2i} = 0$) have a zero on the unit circle that no period can move. By (7), the deviation along such a zero keeps a constant magnitude (a constant position offset for $c_{1i} = 0$, an alternating velocity for $c_{2i} = 0$), so these zeros do not support a destructive ZDA.

4.2. Failure of Fixed-Schedule Defense Against Attacker A3

Proposition 1: Suppose agent i with $c_{1i}c_{2i} < 0$ uses a fixed period T_i, for example the asynchronous base period T/f_i ($f_i \in \mathbb{Z}^+$, $f_i > 1$) of [1]. If the attacker knows T_i and starts from $e_i(k_0) = \xi_0 w_i$, the input $a_i(k) = a^\star_i(T_i, \xi(k))$ keeps $C_i e_i(k) = 0$ for all $k \geq k_0$ while $|\xi(k)| = |\lambda_0^{(i)}(T_i)|^{k-k_0}|\xi_0|$ grows without bound. The residual then equals the measurement noise, and the detection probability equals the false-alarm probability.

Proof: Apply (6) and (7) recursively with $T = T_i$.

A deterministic schedule can be recovered from M consecutive timestamps; in Section VII, attacker A4 recovers it after one interval.

Figure 2 shows the zero locus for the two agent types. Under a deterministic schedule, agents 3 and 4 offer one fixed target; under ARAS the target alternates between the two marked values.

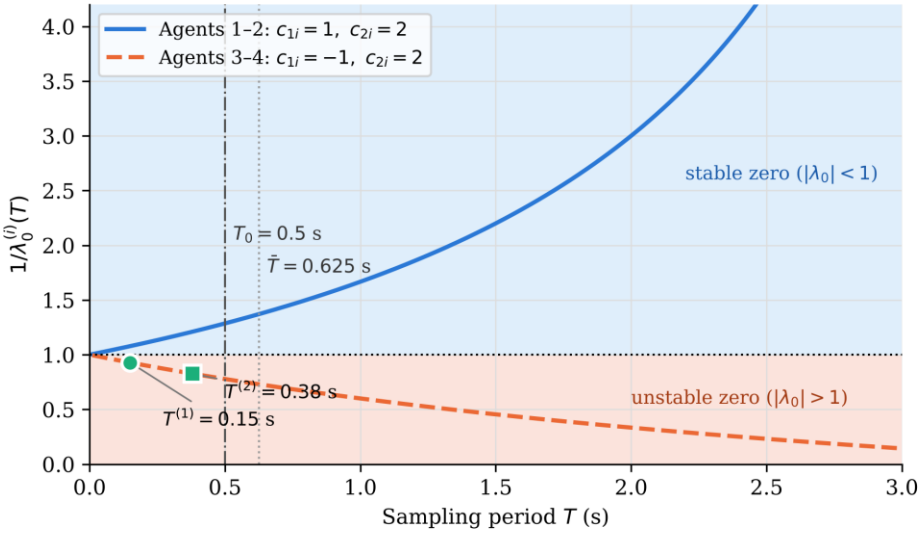


Figure 2. Reciprocal of the sampling zero, $1/\lambda_0^{(i)}(T)$, versus T. Values below 1 correspond to unstable zeros (shaded red). Agents 1 and 2 ($c_{1i}c_{2i} > 0$) have stable zeros for every T. Agents 3 and 4 ($c_{1i}c_{2i} < 0$) have unstable zeros for every T; the two ARAS modes give the distinct zeros $\lambda_0(0.15) = 1.078$ and $\lambda_0(0.38) = 1.210$, so no single zero is common to both modes. $T_0 = 0.5$ s is the nominal period and $\bar{T} = 0.625$ s the consensus bound.

5. Aras Defense Mechanism

5.1. Secure Mode Set Design

Every protected agent has a small number of allowed choices of sampling periods $\mathcal{T}_i = \{T_i^{(1)}, \ldots, T_i^{(m)}\}$.

Definition 1 (Secure mode set): $\mathcal{T}_i$ is secure if the intersection of unstable zeros across all modes is empty:

$$Z_i^u \triangleq \cap_{r=1}^{m} \{\, \lambda_0^{(i)}(T_i^{(r)}) : |\lambda_0^{(i)}(T_i^{(r)})| > 1 \,\} = \emptyset. \qquad (8)$$

Constructive design. By Theorem 1(iii), $T \mapsto \lambda_0^{(i)}(T)$ is injective on the admissible periods, so any $\mathcal{T}_i$ with at least two distinct periods satisfying Assumption 3 is secure.

Definition 2 (Network-level security): For a mode profile $\sigma = (r_1, \ldots, r_N)$, let $Z(\sigma) = \cup_i \{\lambda_0^{(i)}(T_i^{(ri)}) : |\lambda_0^{(i)}(T_i^{(ri)})| > 1\}$. The network common unstable set is $Z_net = \cap_\sigma Z(\sigma)$, the intersection over all profiles in $\mathcal{T}_1 \times \cdots \times \mathcal{T}_N$.

Lemma 2: $Z_net = \cup_i Z_i^u$. Hence, if every agent has a secure mode set, $Z_net = \emptyset$.

Proof: If $\lambda \in Z_i^u$, then $\lambda \in Z(\sigma)$ for every σ, so $\cup_i Z_i^u \subseteq Z_net$. Conversely, if $\lambda \notin Z_i^u$ for every i, pick for each agent a mode r_i whose unstable zero differs from λ (or that has none). The profile $\sigma = (r_1, \ldots, r_N)$ gives $\lambda \notin Z(\sigma)$, so $\lambda \notin Z_net$. ■ Since every agent draws its modes independently with probabilities at least p_min, every profile occurs with probability at least p_min^N at each round, so the condition of Lemma 2 is the one that holds along the realized schedule.

Detectability margin. Security in the sense of (8) only states that the zeros differ. How much they differ sets the size of the residual. By (6), two modes require output-nulling inputs that differ by $|\xi|\delta_i$, where

$$\delta_i = 2c_{1i}^2 \min_\{r \neq s\} \left| 1/(T_i^{(r)}c_{1i} + 2c_{2i}) - 1/(T_i^{(s)}c_{1i} + 2c_{2i}) \right|. \qquad (9)$$

A good mode set keeps every period below $\bar{T}$, spreads the periods to enlarge δ_i, and avoids very short periods, which raise the message rate. For $c_{1i} = -1$, $c_{2i} = 2$ and $\mathcal{T}_i = \{0.15, 0.38\}$ s, $\delta_i = 0.033$ and $g_i(T) \in \{0.289, 0.688\}$, so a wrong guess by the attacker produces an output of only 1% to 2.3% of $|\xi|$. Detection therefore relies on the growth of ξ, which explains the detection delays reported in Section VII.

5.2. Adaptive Randomized Sampling Law

At each sampling instant t_k^i, a protected agent i draws its period $T_i(k) \in \mathcal{T}_i$ from the probability vector $p_i(k) = [p_{i1}(k), \ldots, p_{im}(k)]^\top$ using its private generator. The vector lies in $\Delta_min = \{p : \Sigma_r p_r = 1,\ p_r \geq p_min\}$ with $0 < p_min \leq 1/m$. Let $s_i(k)$ be the detector statistic (13) and τ_i its threshold. The risk index

$$\varrho_i(k) = \min\{ 1, \max\{ 0, (s_i(k) - W)/(\tau_i - W) \} \} \quad (10)$$

is 0 when the statistic is at or below its attack-free mean W and 1 at the alarm threshold. The probabilities are updated as

$$p_i(k) = \Pi_{\Delta\min}[\, p_i(k-1) + \eta_1 \varrho_i(k)(1_m/m - p_i(k-1)) - \eta_2(1 - \varrho_i(k))(\varepsilon_i - \bar{\varepsilon_i} 1_m) \,], \quad (11)$$

where $\Pi_{\Delta\min}$ is the Euclidean projection onto $\Delta_{\min}$ (computed by sorting, O(m log m)), $\varepsilon_{ir} = T_0/T_i^{(r)}$ is the number of samples per nominal period in mode r, $\bar{\varepsilon_i} = (1/m)\Sigma_r \varepsilon_{ir}$, and $\eta_1, \eta_2 > 0$. The first term moves the distribution toward uniform, which maximizes the probability $1 - \max_r p_{ir}$ that the attacker guesses the current mode wrongly (Theorem 4). The second term moves weight toward long periods while the residual is quiet, which saves messages. With $\varrho_i = 0$ the iteration settles at the vertex of $\Delta_{\min}$ that puts $1 - (m-1)p_{\min}$ on the longest period and $p_{\min}$ on the others; for m = 2 and $p_{\min} = 0.30$ this is (0.30, 0.70). On an alarm, p_i is reset to uniform. The floor $p_{\min}$ keeps the schedule random; with $p_{\min} = 0$ the quiet-state schedule becomes deterministic and Proposition 1 applies again (Section VII-E). The attacker is allowed to know $p_i(k)$ (class A3); it does not know the draw.

5.3. Distributed Windowed Detector

Agent i knows the command $u_i(k)$ it sent and the period $T_i(k)$ it used, and it propagates the predictor

$$\hat{z}_i(k+1) = \Phi(T_i(k))\, \hat{z}_i(k) + \Gamma(T_i(k))\, u_i(k), \quad \hat{y}_i(k) = C_i \hat{z}_i(k), \quad (12)$$

initialized with the known deployment state. By (5), the residual is $r_i(k) = C_i e_i(k) + n_i(k)$. The agent computes the windowed χ^2 statistic

$$s_i(k) = \Sigma_{\ell=k-W+1}^{k} r_i^2(\ell)/\sigma_i^2, \quad r_i(\ell) = \tilde{y}_i(\ell) - \hat{y}_i(\ell). \quad (13)$$

Without attack, $e_i \equiv 0$ and the residuals are i.i.d. $N(0, \sigma_i^2)$ for any schedule, because the draws are independent of the noise; hence $s_i(k)$ is exactly χ^2_W distributed for $k \geq W$. If the initial state is uncertain, a Luenberger correction can be added to (12); it does not affect attacks with $C_i e_i \equiv 0$ and only changes the residual covariance.

An alarm is raised if $s_i(k) > \tau_i$, where

$$\tau_i = F^{-1}_{\chi^2_W}(1 - \alpha_{FA}) \quad (14)$$

sets the per-sample false-alarm probability α_{FA}. *Threshold selection procedure.* (a) Choose W: a longer window accumulates more of a persistent residual but reacts later; its length in seconds is $W \cdot \bar{T_i}$, with $\bar{T_i}$ the mean period. (b) Choose α_{FA} and set τ_i by (14). (c) Because consecutive windows overlap, alarms are correlated and α_{FA} does not equal the alarm rate in time. We therefore convert α_{FA} into alarms per agent-hour by Monte Carlo runs without attack and pick the largest α_{FA} that meets the operator budget. (d) Verify the miss rate and delay on attack runs (Section VII-D). The detector is distributed in the strict sense: each agent tests only its own residual and sends no detector messages. After an alarm, the agent stops trusting its command path and switches to a fallback channel; in Section VII this removes the attack input.

Algorithm 1 — ARAS at Agent *i*

1: Input: $\mathcal{T}_i$, $p_{\min}$, η_1, η_2, W, τ_i, T_0
2: $p_i(-1) \leftarrow (1/m)\cdot 1_m$; $\hat{z}_i(0) \leftarrow z_i(0)$
3: for k = 0, 1, 2, … do
4: Measure $\tilde{y}_i(k)$; $r_i(k) \leftarrow \tilde{y}_i(k) - C_i\hat{z}_i(k)$; $s_i(k)$ by (13)
5: if $s_i(k) > \tau_i$ then raise alarm; isolate command path;
6: $p_i(k) \leftarrow (1/m)\cdot 1_m$
7: else $p_i(k)$ by (10) and (11)
8: Draw $T_i(k) \sim p_i(k)$ from the private generator
9: Receive neighbor states; compute $u_i(k)$ by (2); send command
10: $\hat{z}_i(k+1) \leftarrow \Phi(T_i(k))\hat{z}_i(k) + \Gamma(T_i(k))u_i(k)$
11: Wait $T_i(k)$
12: end for

6. Stability and Security Analysis

6.1. No-Common-Unstable-Zero Theorem

Theorem 2 (No persistent perfectly stealthy *ZDA):* Let Assumptions 3 to 6 hold, let agent $i \in A_c$ have $c_{1i}c_{2i} < 0$ and a mode set with $m \geq 2$, and let $a_i(k)$ be $\mathcal{J}_i(k)$-measurable (classes A1 to A4; the attacker may also randomize, independently of the draw). Suppose $C_i e_i(k_0) = 0$ and $e_i(k_0) \neq 0$. Then for every $K \geq 1$,

$$\Pr\{ C_i e_i(k_0+1) = \cdots = C_i e_i(k_0+K) = 0 \mid \mathcal{J}_i(k_0) \} \leq (1 - (m-1)p_min)^K. \quad (15)$$

In particular, a perfectly stealthy attack persists forever with probability zero.

Proof: On the event $C_i e_i(k) = 0$, $e_i(k) \neq 0$, we have $e_i(k) = \xi(k) w_i$ with $\xi(k) \neq 0$. By (6), $C_i e_i(k+1) = 0$ holds if and only if $a_i(k) = a^\star_i(T_i(k), \xi(k))$. Since $c_{1i} \neq 0$, the map $T \mapsto 1/(Tc_{1i} + 2c_{2i})$ is injective, so the m candidate values $a^\star_i(T_i^{(r)}, \xi(k))$ are distinct and $a_i(k)$ can equal at most one of them. The draw $T_i(k)$ is independent of $\mathcal{J}_i(k)$ and of the attacker's own randomness, so the conditional probability of nulling the next output is at most $\max_r p_{ir}(k) \leq 1 - (m-1)p_min$. If the output is nulled, (7) gives $e_i(k+1) = \lambda_0^{(i)} \xi(k) w_i \neq 0$, and $\mathcal{J}_i(k) \subseteq \mathcal{J}_i(k+1)$. Applying the same bound at $k_0 + 1$, …, $k_0 + K$ and the tower property of conditional expectation yields (15).

Corollary 1 (Exponential attacks, network level): A geometric attack $a_i(k) = a_0 \lambda^k$ with $|\lambda| > 1$ is perfectly stealthy on all realized steps only if λ is an unstable zero of every realized mode, that is, $\lambda \in Z_i^u$. Under Definition 1 this set is empty, and by Lemma 2 the same holds for the network: no λ is an unstable zero of every realized profile. This is the statement of the original version of the theorem; Theorem 2 extends it to attacks with arbitrary, state-dependent amplitudes.

Proposition 3 (The information pattern is necessary): If the attacker knows $T_i(k)$ before it commits $a_i(k)$ (class A5), the input $a_i(k) = a^\star_i(T_i(k), \xi(k))$ keeps $C_i e_i \equiv 0$ and gives $|\xi(k)| \geq (\min_r |\lambda_0^{(i)}(T_i^{(r)})|)^{k-k_0} |\xi(k_0)|$, which diverges for a mode set of unstable zeros. If the current period leaks with probability q at each step, (15) becomes $(1 - (1-q)(m-1)p_min)^K$.

Proof: $\ker C_i = \mathrm{span}\{w_i\}$ does not depend on the period, so (6) and (7) can be applied with the known period at every step. The bound with leaks follows from the proof of Theorem 2, since a leak occurs with probability q independently of the draw.

Remark 2 (Partial and delayed knowledge): Theorem 2 already lets the attacker know $p_i(k)$ and every past period. Since the draws are conditionally independent given $p_i(k)$, past periods carry no information about $T_i(k)$ beyond $p_i(k)$, and delayed or estimated knowledge of p_i (class A4) can only weaken the attacker. The bounds therefore remain valid when the attacker's knowledge of the schedule is partial or delayed. Knowledge of the current draw is a different matter: by Proposition 3 the guarantee degrades with the leak probability q and vanishes at $q = 1$. The seed of the generator must be protected like a cryptographic key.

6.2. Mean-Square Practical Consensus

Because the agents sample at different random instants, (3) does not describe the closed loop. We index the network sampling events (the union of all agents' instants) by ℓ, with inter-event time Δ_ℓ and updating agent i_ℓ. Let $Q \in \mathbb{R}^{N \times (N-1)}$ have orthonormal columns spanning $1_N^\perp$, and define the disagreement state $\zeta = [(Q^\top x)^\top, (Q^\top v)^\top, u^\top]^\top$, where u collects the held commands. Without attack, the closed loop is

$$\zeta(\ell+1) = J_{i_\ell}\, F(\Delta_\ell)\, \zeta(\ell) + G_{i_\ell}\, \nu(\ell), \quad (16)$$

where $F(\Delta)$ propagates the double integrators over Δ with held commands, J_i replaces u_i by the protocol (2), and $\nu(\ell)$ collects the navigation noise used in that update. Since $L1_N = 0$, (2) depends on x and v only through $Q^\top x$ and $Q^\top v$, so (16) is closed in ζ.

Assumption 7 (Windowed contraction): There exist $P \succ 0$, $\varrho \in (0, 1)$, and $T_w > 0$ such that, for every admissible schedule (any periods from the mode sets, in any order and phase) and every time window of length T_w, the product Ψ of the maps $J_{i_\ell} F(\Delta_\ell)$ over the events in the window satisfies $\Psi^\top P \Psi \preceq \varrho P$.

Theorem 3 (Robust practical consensus): Under Assumptions 1 to 5 and 7 and without attack, for any adaptation of the probabilities with $p_i(k) \in \Delta_min$, possibly depending on past residuals,

$$\mathbb{E}\{\|\zeta(t)\|^2_P\} \leq \varrho^{\lfloor t/T_w \rfloor}\, \mathbb{E}\{\|\zeta(0)\|^2_P\} + \gamma \sigma_\nu^2/(1-\varrho), \quad (17)$$

with $\gamma = \bar{n}\, \bar{g}$, where $\bar{n}$ bounds the number of events in a window and $\bar{g}$ bounds $\|P^{1/2} \Psi' G\|^2$ over partial products Ψ'. The consensus error $e = (L \otimes I)\, z$ satisfies $\limsup \mathbb{E}\{\|e(t)\|^2\} \leq c_0 \gamma \sigma_\nu^2/(1-\varrho)$ for a constant c_0 that depends on L and P.

Proof: Over one window, $\zeta^{+} = \Psi\zeta + \Sigma_j \Psi_j' G_{i_j} \nu_j$. The schedule depends only on the draws and on the measurement noise n_i through the residuals, which are independent of ν (Assumption 4). Conditioning on the schedule, the cross terms vanish because the ν_j are zero mean and independent of ζ, and $E\{\|\zeta^{+}\|^2 _P\} \le \varrho E\{\|\zeta\|^2 _P\} + n\bar{g}\sigma_\nu^2$ by Assumption 7. Since $n \le \Sigma_i(\lceil T_w/\min_r T_i^{(r)} \rceil + 1)$ and the matrices are bounded on the compact set of inter-event times, γ is finite. Iterating over $\lfloor t/T_w \rfloor$ windows gives (17), and $\|e\| \le c_0^{1/2}\|\zeta\|_P$ gives the second claim.

Verification for Section VII. With P = I and T_w = 3 s, we sampled 9 000 schedules of the four-agent example (agents 1 and 2 at 0.5 s, agents 3 and 4 drawing from {0.15, 0.38} s with three different distributions) and evaluated $\|\Psi\|_2$ over a 3 s window. The largest value was 0.553, so Assumption 7 holds on these samples with $\varrho = 0.31$. This is numerical evidence, not a certificate; a certificate can be obtained by gridding Δ and solving a linear matrix inequality. The individual modes satisfy $T < \bar{T} = 0.625$ s from Section III-D.

Remark 3 (Damped agents): For $d_i > 0$ the single-agent matrices change to $\Phi = [1, (1 - e^{-d_i T})/d_i; 0, e^{-d_i T}]$ and $\Gamma = b_i[T/d_i - (1 - e^{-d_i T})/d_i^2; (1 - e^{-d_i T})/d_i]$. The continuous-time zero $-c_{1i}/c_{2i}$ is unchanged, but (4) no longer holds in closed form. The proof of Theorem 2 only uses the fact that the output-nulling inputs of different modes differ, which can be checked numerically from $a^{\star}_i = -C_i\Phi(T)e_i/(C_i\Gamma(T))$; Section VII-F tests random $d_i \in [0, 0.3]$ and $b_i \in [0.7, 1.3]$.

6.3. Detectability of Imperfect and Schedule-Aware Attacks

Theorem *4 (Residual lower bound):* Under the conditions of Theorem 2, let $e_i(k) = \xi(k)w_i$ with $\xi(k) \neq 0$. For any $\mathcal{I}_i(k)$-measurable $a_i(k)$,

$$\Pr\{\, |C_i e_i(k+1)| \ge \tfrac{1}{2}\, \bar{g}_i\, \delta_i\, |\xi(k)| \;\mid\; \mathcal{I}_i(k) \,\} \ge 1 - \max_r p_{ir}(k) \ge (m-1)p_min, \tag{18}$$

where $\bar{g}_i = \min_r |g_i(T_i^{(r)})|$ and δ_i is given by (9).

Proof: The candidates $a^{\star}_i(T_i^{(r)}, \xi)$ are at least $\delta_i|\xi|$ apart. By the triangle inequality, $|a - a^{\star}_i(T_i^{(r)}, \xi)| < \delta_i|\xi|/2$ can hold for at most one r. For every other mode, (6) gives $|C_i e_i(k+1)| \ge \bar{g}_i\delta_i|\xi|/2$, and the draw falls on one of these modes with conditional probability at least $1 - \max_r p_{ir}(k)$.

Corollary *2 (Detection over a window):* If $|\xi| \ge \xi_$ on W consecutive steps, the number J of steps with $|C_i e_i| \ge \tfrac{1}{2}\bar{g}_i\delta_i\xi_$ stochastically dominates a Binomial(W, (m − 1)p_min) variable. Given J = j, s_i dominates a noncentral χ^2_W variable with noncentrality $j(\bar{g}_i\delta_i\xi_)^2/(4\sigma_i^2)$. Hence $\Pr\{\text{alarm}\} \ge \Sigma_j \Pr\{J = j\} \cdot \Pr\{\chi'^2_W(j(\bar{g}_i\delta_i\xi_)^2/(4\sigma_i^2)) > \tau_i\}$. Attackers A1 and A2 produce additional residual from their model mismatch; A1 is mismatched at every step, because no ARAS mode equals T_0.

The bound replaces the heuristic first-order estimate of the previous version. It is conservative, but it shows the main mechanism: the residual is proportional to the deviation $|\xi|$, so detection follows the exponential growth of the attack, and the margin $\bar{g}_i\delta_i/2$ together with the noise level σ_i fixes how far the attack can progress before it becomes visible.

6.4. Trade-Offs Between Randomization, Detection, Consensus, and Communication

Let $\bar{T}_i = \Sigma_r p_{ir} T_i^{(r)}$ be the mean period of agent i. Because the periods are drawn independently, the renewal theorem gives a long-run update rate $1/\bar{T}_i$, and the overhead of agent i relative to the nominal period is

$$O_i = T_0/\bar{T}_i - 1. \tag{19}$$

Each update costs deg(i) messages, so the network overhead is the degree-weighted average of O_i. The probability that a schedule-aware attacker misses the current mode is $\mu_i = 1 - \max_r p_{ir}$. For two modes, with weight $p \le 1/2$ on the short one, both O_i and $\mu_i = p$ increase with p: every gain in unpredictability costs messages, and the uniform distribution gives the most protection at the highest cost. In the quiet state of (11), p = p_min, so p_min sets the operating point. For $\mathcal{T}$ = {0.15, 0.38} s and p_min = 0.30, $\bar{T}_i = 0.311$ s, $O_i = 60.8\%$ for each protected agent, and the degree-weighted network overhead is 30.4%; the simulated value is 29.7%. Shorter mean periods also tighten the consensus loop, since each mode contracts faster, but the consensus error is dominated by the navigation noise (Table III). Detection improves with μ_i through Corollary 2, and the detector itself adds no messages. Fig. 6(a) shows the resulting curve.

Remark 4 (Communication overhead): The average number of updates of agent i per nominal period is $\bar{f}_i = T_0/\bar{T}_i$, and the overhead with respect to the synchronous baseline (one sample per nominal period) is $\bar{f}_i - 1$.

Because the energy term in (11) favors long periods when risk is low, overhead can be lower than with a fixed high-rate schedule. In Section VII, ARAS adds 29.7% messages at the network level, deterministic asynchronous sampling adds 49.0%, and uniform randomization adds 43.3%.

6.5. Computational and Communication Cost

Per sampling instant, a protected agent updates the predictor (12) with a 2 × 2 matrix-vector product (about 10 floating-point operations), updates the window sum of (13) with a running total (3 operations and a W-entry buffer), evaluates (10) (3 operations), performs the update (11) and its projection (about 8m operations plus a sort of m numbers), and draws one random number. For m = 2 and W = 20 this is about 40 operations and 24 stored numbers per update. An interpreted Python implementation took 2.4 μs per update on one core of a 2.1 GHz Xeon processor, which is negligible against sampling periods of 0.15 s or more. The detector sends no messages. The only communication cost of ARAS is the higher update rate (19); dual-rate sampling (S2) instead raises the sensing rate. Table III reports both.

7. Simulations

7.1. Setup

We replicate the quadrotor UAV formation of [1], [9]: N = 4 agents with outer-loop second-order dynamics (1), $\alpha = 1$, $\beta = 0.8$. The Laplacian is L = [3 −1 −1 −1; −1 2 −1 0; −1 −1 3 −1; −1 0 −1 2], whose eigenvalues are 0, 2, 4, 4. The consensus bound of Section III-D gives $\bar{T} = \min\{2\beta/\alpha, 2/(\beta\mu_N)\} = 0.625$ s, and the nominal period is $T_0 = 0.5$ s.

Heterogeneous outputs. Agents 1–2: $c_{1i} = 1$, $c_{2i} = 2$ ($c_{1i}c_{2i} > 0$, stable zeros $\lambda_0 \approx 0.78$ at T = 0.5). Agents 3–4: $c_{1i} = -1$, $c_{2i} = 2$ ($c_{1i}c_{2i} < 0$, unstable zero $\lambda_0 = 1.29$ at T = 0.5). The attacker compromises the command packets of agents 2 and 3. Agent 2 has a stable zero and offers no destructive ZDA, so the attack runs through agent 3.

ARAS parameters. $\mathcal{T}_3 = \mathcal{T}_4 = \{0.15, 0.38\}$ s, p_min = 0.30, $\eta_1 = 0.2$, $\eta_2 = 0.05$, W = 20, $\alpha_FA = 10^{-4}$ ($\tau = 52.4$). Agents 1 and 2 sample at T_0. Zero verification: $\lambda_0(0.15) = 1.078$, $\lambda_0(0.38) = 1.210$; since $1.078 \neq 1.21$, $Z_3^u = Z_4^u = \emptyset$ (Definition 1 and Theorem 1(iii)). The value p_min = 0.30 is the smallest floor on the grid of Fig. 6(a) whose upper 95% confidence limit on the attack success rate stays below 10%.

Noise, attack, and success criterion. Measurement noise $\sigma_i = 0.002$, navigation noise $\sigma_\nu = 0.005$, initial positions uniform in [−2, 2] m and velocities in [−1, 1] m/s. The attack starts at t = 6 s, after the formation has converged, with an initial deviation $\xi_0 = 0.02$ along w_3 (the standard initial condition of a ZDA [4]; in practice a single sub-noise kick achieves it), and runs to the horizon of 25 s. The consensus error is $E(t) = \max_i |(Lx)_i(t)|$; its 99th percentile without attack is 0.036 m. An attack counts as successful if E(t) exceeds E_dmg = 1 m before the first alarm raised after the attack start. A run with no alarm before damage is a miss. The attack success rate (ASR) is the fraction of successful runs; we report it with Wilson 95% confidence intervals [34]. Each cell uses 500 independent trials (different initial states, noise, and draws). For A3 and A4 we run both the MAP and the hedging rule and report the one with the higher ASR. Message overhead is measured in attack-free runs relative to S1. After an alarm, the attack input is removed (Section V-C).

Strategies. S1: synchronous sampling at T_0 [2]. S2: dual-rate sampling [8], with commands held over T_0 and outputs of agents 3 and 4 sampled every $T_0/2$. S3: deterministic asynchronous sampling [1], agents 3 and 4 at $T_0/2$. S4: randomized sampling over $\mathcal{T}_i$ with uniform, fixed probabilities. S5: ARAS. S6: synchronous sampling with an additive Gaussian watermark (standard deviation 0.05) on the commands of agents 3 and 4 [15]. All strategies use the same detector, threshold, noise, graph, and attacker.

The zero locus $1/\lambda_0$ versus the sampling period for both types of agents is presented in Fig. 2, including both modes of ARAS. The fixed synchronous zero for agents 3–4 at T = 0.5 ($1/\lambda_0 = 0.78$, inside the band of unstable zeros) is the attacker's static target; ARAS randomizes this target between two values.

7.2. Consensus Error Under Attack

Fig. 3 shows E(t) for one trial (seed 3) without attack and under attacker A3 with S1, S3, and S5. The attack starts at t = 6 s. Under S1 and S3 the attacker knows the period, the residual stays at the noise level, and E(t)

crosses 1 m at about 18 s. Under ARAS the deviation grows for a while, the detector of agent 3 raises an alarm at t = 15.0 s, the command path is isolated, and the formation returns to its nominal error.

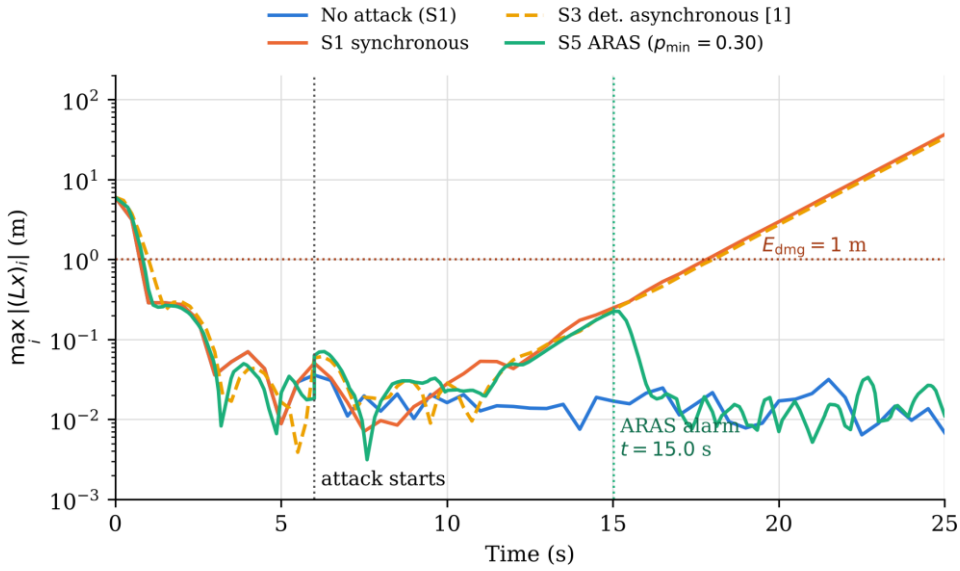


Figure 3. Consensus error $\max_i|(Lx)_i|$ on a logarithmic scale for one trial under the schedule-aware attacker A3 (attack from t = 6 s). S1 and S3 diverge without an alarm. ARAS (S5) raises an alarm at t = 15.0 s, before the damage threshold E_dmg = 1 m is reached.

7.3. Comparative Attack Success Rate

Fig. 4 and Table II give the ASR of the six strategies against the four attacker classes, and Table III gives detection, cost, and consensus quality under A3.

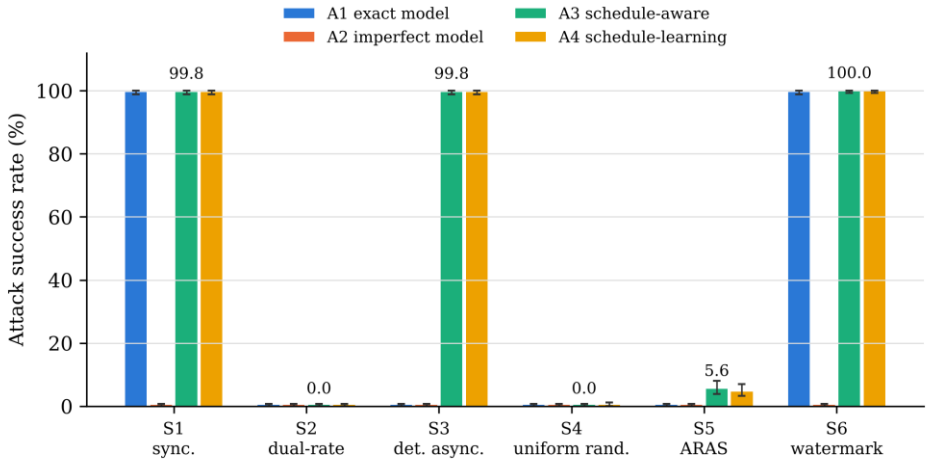


Figure 4. Attack success rate (500 trials per bar, Wilson 95% intervals) for six strategies and four attacker classes. Labels give the value for the schedule-aware attacker A3.

Table 2. Attack Success Rate (%) with 95% Confidence Intervals (N = 4, 500 Trials Per Cell)

Strategy	A1	A2	A3	A4
S1: Synchronous [2]	99.8 [98.9, 100.0]	0.0 [0.0, 0.8]	99.8 [98.9, 100.0]	99.8 [98.9, 100.0]
S2: Dual-rate [8]	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]
S3: Det. asynchronous [1]	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	99.8 [98.9, 100.0]	99.8 [98.9, 100.0]
S4: Randomized, uniform	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	0.2 [0.0, 1.1]

S5: ARAS (proposed)	0.0 [0.0, 0.8]	0.0 [0.0, 0.8]	5.6 [3.9, 8.0]	4.8 [3.2, 7.0]
S6: Watermarking [15]	99.8 [98.9, 100.0]	0.0 [0.0, 0.8]	100.0 [99.2, 100.0]	100.0 [99.2, 100.0]

A1: exact model, nominal period. A2: schedule-aware, 5% coefficient error. A3: schedule-aware, knows $p_i(k)$. A4: learns p_i from the last 20 periods. Success: E(t) > 1 m before the first alarm.

Table 3. Detection, Cost, and Consensus Quality (ATTACKER A3, N = 4)

Strategy	Det. (%)	Delay (s)	Msg. ovh. (%)	Sens. ovh. (%)	RMS err. (cm)	Settle (s)	FA runs (%)
S1: Synchronous [2]	0.2	—	0.0	0.0	1.73	2.61	0.5
S2: Dual-rate [8]	100.0	8.6 ± 0.0	0.0	49.0	1.78	2.73	0.5
S3: Det. asynchronous [1]	0.4	—	49.0	49.0	1.47	2.72	1.0
S4: Randomized, uniform	100.0	7.7 ± 0.1	43.3	43.3	1.57	2.76	0.5
S5: ARAS (proposed)	100.0	9.9 ± 0.1	29.7	29.7	1.59	2.72	1.0
S6: Watermarking [15]	0.0	—	0.0	0.0	3.62	3.15	0.0

Det.: runs with an alarm after the attack start. Delay: mean time from attack start to first alarm ± 95% half-width (shown when more than 5% of runs raised an alarm). Overheads: control messages and sensor samples per second relative to S1, attack-free. RMS err.: root-mean-square of E(t) over 6 to 25 s, attack-free. Settle: last time E(t) ≥ 0.1 m. FA runs: attack-free runs with at least one alarm in 25 s (200 runs).

Three results stand out. First, deterministic asynchronous sampling [1] stops the exact-model attacker A1 (0% ASR) but not the schedule-aware attacker A3 (99.8%), in line with Proposition 1. A4 needs one observed interval to learn the fixed period, and does as well as A3. Second, ARAS reduces the ASR under A3 to 5.6 [3.9, 8.0]% and under A4 to 4.8 [3.2, 7.0]%, while A1 and A2 are always detected because their inputs never match a realized mode. The ARAS alarms come 9.9 s after the attack start on average, which follows from Corollary 2: the deviation must grow by roughly two orders of magnitude before the mismatch residual stands out of the noise. Third, ARAS is not the most secure scheme in this table. Uniform randomization (S4) and dual-rate sampling (S2) reached 0% ASR. S4 pays 43.3% extra messages against 29.7% for ARAS. S2 sends no extra messages but takes 49.0% more sensor samples, all of them intersample outputs (Section VIII). Watermarking (S6) gives no protection against A1, A3, or A4, as expected for a ZDA with an accurate model, and doubles the consensus error. Relative to S3, ARAS lowers the ASR under A3 and cuts the message overhead from 49.0% to 29.7%, a 39% reduction.

The new results replace those of the previous version, which used 30 trials, a damage criterion of five times the nominal error, and a different noise setting. Under the five-times criterion (0.18 m), no strategy detected the attack before damage, because the residual of a mismatched step is far below the noise when the deviation is that small; we adopted a fixed 1 m threshold, which corresponds to a loss of formation for the quadrotors of [9], and report the sensitivity to this choice in Table VI. The previous overhead of 27% was not consistent with the mode set {0.15, 0.38} s, whose minimum per-agent overhead is 31.6%; the values reported here are measured.

7.4. Detector Design: False Alarms, Misses, and Delay

Table IV and Fig. 5 apply the threshold procedure of Section V-C to ARAS under A3. For each window length W and per-sample false-alarm probability α_FA, we measured the alarm rate in 40 attack-free runs of 300 s and the ASR, detection rate, and delay in 200 attack runs per rule. For W = 20, lowering α_FA from 10^{-2} to 10^{-4} cuts the false alarms from 21.4 to 0.39 per agent-hour while the ASR rises from 1.5% to 5.0%; at 10^{-5} and below, no false alarm occurred in about 13 agent-hours of operation per setting. The window length matters less than the threshold: W = 10 reacts slightly faster, W = 40 is slightly less sensitive at the same α_FA. We use W = 20 and α_FA = 10^{-4}, which gives about 0.4 false alarms per agent-hour. All attack runs eventually raised an alarm within the horizon, so the false-negative rate within 19 s is zero; the relevant miss rate is the fraction of runs in which the alarm came after the damage threshold, which is the ASR.

Table 4. Detector Operating Points for Aras Under Attacker A3

W	α_FA	τ	FA per agent-h	Miss before damage (%)	No alarm (%)	Delay (s)
10	1e−2	23.2	31.11	1.5	0.0	7.72
10	1e−3	29.6	4.11	3.0	0.0	9.12
10	1e−4	35.6	0.23	4.5	0.0	9.67
10	1e−5	41.3	0.00	5.0	0.0	9.96
10	1e−6	46.9	0.00	6.0	0.0	10.13
20	1e−2	37.6	21.41	1.5	0.0	8.20
20	1e−3	45.3	3.49	4.5	0.0	9.45
20	1e−4	52.4	0.39	5.0	0.0	9.90
20	1e−5	59.0	0.00	5.0	0.0	10.07
20	1e−6	65.4	0.00	6.0	0.0	10.25
40	1e−2	63.7	17.07	3.0	0.0	9.28
40	1e−3	73.4	2.33	4.5	0.0	9.82
40	1e−4	82.1	0.31	5.5	0.0	10.13
40	1e−5	90.1	0.00	7.0	0.0	10.50
40	1e−6	97.7	0.00	7.5	0.0	10.65

FA per agent-h: alarms per agent per hour in 40 attack-free runs of 300 s (about 13 agent-hours per row). Miss before damage equals the ASR. No alarm: attack runs without any alarm in the 19 s after the attack start. 200 trials per rule; the worse rule is reported.

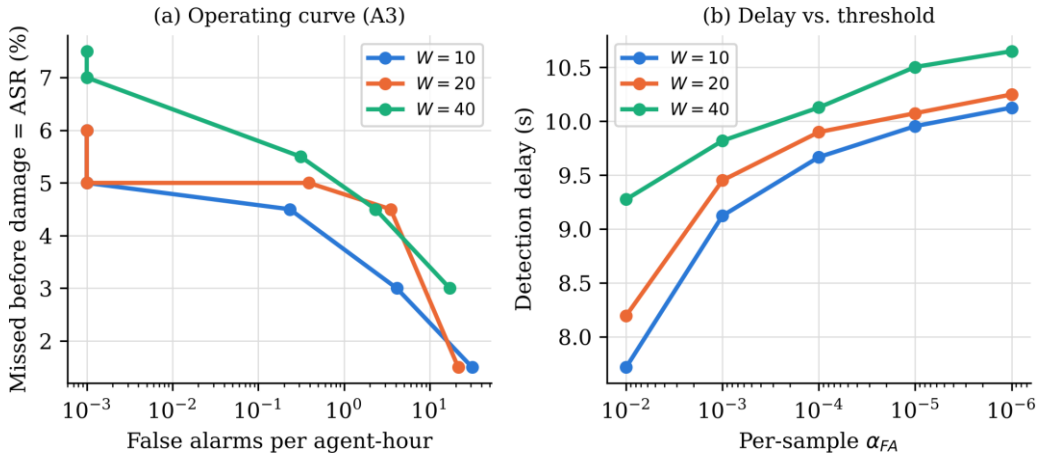


Figure 5. Detector operating curves for ARAS under A3. (a) Fraction of attacks missed before damage versus false alarms per agent-hour, for three window lengths (false-alarm rates of zero are drawn at 10^{-3}). (b) Mean detection delay versus the per-sample false-alarm probability.

7.5. Ablation Study

Table V separates the three components of ARAS: randomization of the period, adaptation of the distribution, and the detector. Removing randomization while keeping the two modes (deterministic cycling 0.15, 0.38, 0.15, …) lets the attacker predict every period, and the ASR rises to 99.0%. Removing the floor (p_min = 0) lets (11) drive the quiet-state schedule to the long mode, and the ASR rises to 94.0%, as Proposition 1 predicts. Removing the detector leaves the attack undetected in every run: randomization creates a residual, but only a test turns it into a decision. The adaptive part matters less. Fixing the distribution at the quiet-state value (0.30, 0.70) raises the ASR from 5.6% to 7.6% at the same message rate, and removing only the residual feedback gives 6.8%; the confidence intervals overlap. The uniform distribution gives the lowest ASR at the highest cost. In short, the floor and the detector carry most of the protection, and the adaptation adds a small gain at no extra cost.

Table 5. Ablation of aras under attacker A3 (N = 4, 500 TRIALS)

Variant	ASR (%) [95% CI]	Det. (%)	Delay (s)	Msg. ovh. (%)
ARAS (full)	5.6 [3.9, 8.0]	100.0	9.9	29.7
No randomization (cycle the two modes)	99.0 [97.7, 99.6]	1.4	—	43.1
No floor (p_min = 0)	94.0 [91.6, 95.8]	11.0	12.2	16.7
No detector	100.0 [99.2, 100.0]	0.0	—	29.7
No adaptation (fixed 0.30/0.70)	7.6 [5.6, 10.3]	100.0	9.9	29.3
No residual feedback ($\eta_1 = 0$)	6.8 [4.9, 9.4]	100.0	9.9	29.7
Uniform, fixed (S4)	0.0 [0.0, 0.8]	100.0	7.7	43.3

7.6. Robustness: Network Size, Topology, Noise, Attack Strength, and Dynamics

Table VI collects the robustness tests, all under attacker A3 with ARAS at p_min = 0.30. For N = 10, 25, and 50 we used random geometric graphs, half of the agents with $c_{1i} = -1$, ⌈0.1N⌉ compromised agents with unstable zeros, 10% message loss, and the Laplacian scaled so that $\mu_N = 4$ (which keeps $\bar{T} = 0.625$ s). The attack started at 20 s to let the larger networks converge, and the horizon was 40 s. For the topology tests (N = 10) we used ring, star, complete, and random geometric graphs. For heterogeneous dynamics we drew five sets of (c_{1i}, c_{2i}, d_i, b_i) with $|c_{1i}| \in [0.5, 1.5]$, $c_{2i} \in [1.5, 2.5]$, $d_i \in [0, 0.3]$, and $b_i \in [0.7, 1.3]$.

The results support five observations. (i) Network size: with 10% message loss, ARAS kept the ASR at 3.5% for N = 4 and 9.0% for N = 10, and no attack succeeded for N = 25 and 50 in 100 trials per rule, while S3 failed in 98% to 99.5% of the runs at every size. Part of the drop for the larger networks comes from the metric: with the scaled Laplacian, the same deviation of one agent gives a smaller E(t), so the attack needs longer to reach 1 m and the detector collects more samples. The network message overhead of ARAS stayed between 28.8% and 30.4%, because each agent randomizes on its own and its cost does not depend on N. (ii) Topology: the ASR of ARAS ranged from 3.5% (star) to 7.0% (complete and random geometric graphs), and S3 failed in 98.5% of the runs on every graph. By (5) the residual of the attacked agent does not depend on the graph, which is why the detection delays were identical across the four graphs; the graph only changes how quickly the deviation appears in E(t). (iii) Heterogeneous dynamics: in three of the five draws ARAS kept the ASR between 7% and 17% while S3 failed in 91% to 100% of the runs. In the other two draws no attack succeeded under either scheme, because the compromised agent had a slow continuous-time zero (small $|c_{1i}|/c_{2i}$) and its deviation did not reach 1 m within the horizon. (iv) Noise and damage threshold: ARAS depends strongly on the signal-to-noise ratio. Its ASR rose from 2.0% at $\sigma_i = 0.001$ to 50.0% at $\sigma_i = 0.005$ and 90.0% at $\sigma_i = 0.01$, and from 5.0% at E_dmg = 1 m to 24.5% at 0.5 m. Uniform randomization degrades later (13.0% and 69.5% at $\sigma_i = 0.005$ and 0.01), and dual-rate sampling later still (35.5% at $\sigma_i = 0.01$). (v) Attack strength: the initial deviation ξ_0 changed the ASR of ARAS only between 1.5% and 5.0%, since the growth rate of the attack is set by the zero, not by ξ_0.

Table 6. Robustness under attacker A3 (ASR IN % WITH 95% CI)

Scenario	S3	S4	S5 ARAS	ARAS msg. ovh. (%)
N = 4, paper graph, 10% loss	99.5 [97.2, 99.9]	0.0 [0.0, 1.9]	3.5 [1.7, 7.0]	29.9
N = 10, RGG, 10% loss	99.0 [96.4, 99.7]	0.0 [0.0, 1.9]	9.0 [5.8, 13.8]	28.8
N = 25, RGG, 10% loss	99.0 [94.6, 99.8]	0.0 [0.0, 3.7]	0.0 [0.0, 3.7]	29.5
N = 50, RGG, 10% loss	98.0 [93.0, 99.4]	0.0 [0.0, 3.7]	0.0 [0.0, 3.7]	30.4
N = 10, ring	98.5 [95.7, 99.5]	—	6.0 [3.5, 10.2]	—
N = 10, star	98.5 [95.7, 99.5]	—	3.5 [1.7, 7.0]	—
N = 10, complete	98.5 [95.7, 99.5]	—	7.0 [4.2, 11.4]	—
N = 10, RGG	98.5 [95.7, 99.5]	—	7.0 [4.2, 11.4]	—

N = 10, heterogeneous dynamics, draw 1	99.0 [94.6, 99.8]	—	7.0 [3.4, 13.7]	—
N = 10, heterogeneous dynamics, draw 2	100.0 [96.3, 100.0]	—	10.0 [5.5, 17.4]	—
N = 10, heterogeneous dynamics, draw 3	0.0 [0.0, 3.7]	—	0.0 [0.0, 3.7]	—
N = 10, heterogeneous dynamics, draw 4	0.0 [0.0, 3.7]	—	0.0 [0.0, 3.7]	—
N = 10, heterogeneous dynamics, draw 5	91.0 [83.8, 95.2]	—	17.0 [10.9, 25.5]	—
N = 4, $\sigma_i = 0.001$	—	0.0 [0.0, 1.9]	2.0 [0.8, 5.0]	—
N = 4, $\sigma_i = 0.002$	—	0.0 [0.0, 1.9]	5.0 [2.7, 9.0]	—
N = 4, $\sigma_i = 0.005$	—	13.0 [9.0, 18.4]	50.0 [43.1, 56.9]	—
N = 4, $\sigma_i = 0.01$	—	69.5 [62.8, 75.5]	90.0 [85.1, 93.4]	—
N = 4, $\xi_0 = 0.005$	—	0.0 [0.0, 1.9]	2.5 [1.1, 5.7]	—
N = 4, $\xi_0 = 0.02$	—	0.0 [0.0, 1.9]	5.0 [2.7, 9.0]	—
N = 4, $\xi_0 = 0.1$	—	0.0 [0.0, 1.9]	1.5 [0.5, 4.3]	—
N = 4, E_dmg = 0.5 m	—	2.5 [1.1, 5.7]	24.5 [19.1, 30.9]	—
N = 4, E_dmg = 1.0 m	—	0.0 [0.0, 1.9]	5.0 [2.7, 9.0]	—
N = 4, E_dmg = 2.0 m	—	0.0 [0.0, 1.9]	1.0 [0.3, 3.6]	—

N = 4, 10: 200 trials per rule; N = 25, 50: 100 trials per rule; topology and noise rows: 200 trials per rule; heterogeneous rows: 100 trials per rule. For $\sigma_i = 0.01$, dual-rate sampling (S2) had ASR 35.5 [29.2, 42.3]%.

7.7. Partial and Delayed Attacker Knowledge

Fig. 6(b) gives the ASR when the current period leaks with probability q (300 trials per point). The ASR of ARAS grows from 6.3% at q = 0 to 27.7% at q = 0.5 and 99.7% at q = 1; uniform randomization stays below 4% up to q = 0.5 and also fails at q = 1, in line with Proposition 3. Delayed or estimated knowledge of the distribution does not help the attacker. For the learning attacker A4 with windows of 5, 20, and 100 periods and delays of 0, 5, and 20 periods, the ASR of ARAS ranged from 2.7% to 6.3% (300 trials each), within the interval of A3. This agrees with Remark 2.

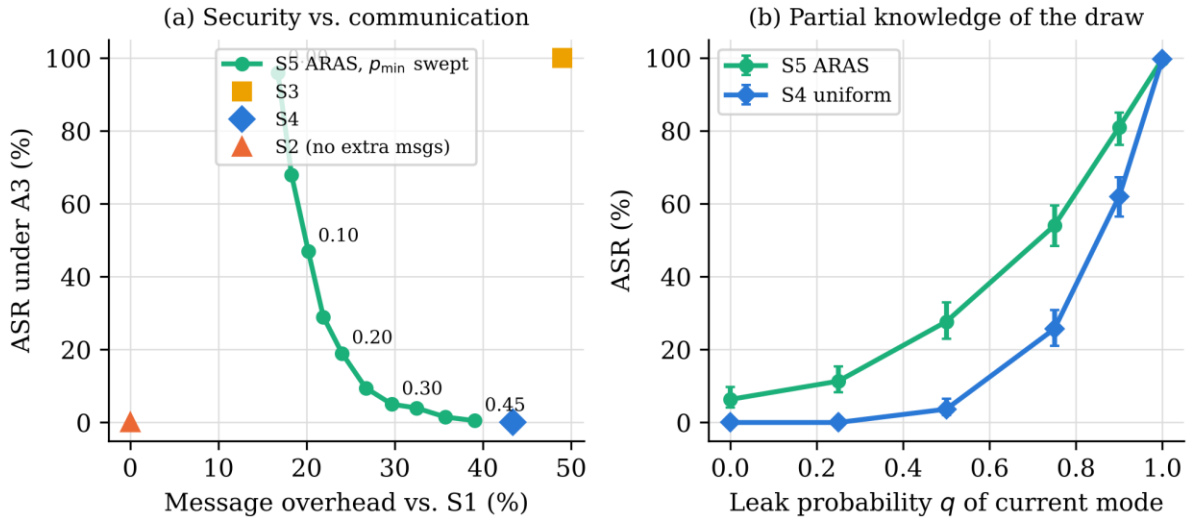


Figure 6. (a) ASR under A3 versus network message overhead as p_min varies from 0 to 0.45 (ARAS), with S2, S3, and S4 for reference. (b) ASR versus the probability q that the current period leaks to the attacker; error bars are Wilson 95% intervals.

7.8. Trade-Off Between Randomization and Overhead

Fig. 6(a) sweeps p_min (200 trials per point). The ASR falls from 96.0% at p_min = 0 to 47.0% at 0.10, 19.0% at 0.20, 5.0% at 0.30, and 0.5% at 0.45, while the network message overhead rises from 16.8% to 39.1%. The RMS consensus error stayed between 1.56 and 1.61 cm over the sweep, so consensus quality hardly constrains the choice. The operator picks p_min from the overhead budget and the acceptable ASR; S3 lies above the curve and S4 at its end.

8. Limitations and Stronger Adversaries

Timestamps. ARAS does not hide timestamps; Theorem 2 lets the attacker read all of them. The scheme needs only that a timestamp reveals a period after the period has started, never before. A protocol that announces the next period in advance, for example in a beacon, turns A3 into A5.

Knowledge of the draw. If the generator seed or state leaks, the attacker becomes clairvoyant and Proposition 3 shows that no amount of randomization helps; Fig. 6(b) shows the gradual loss for partial leaks. The generator should run in a trusted module with a secret seed.

Packet creation and continuous-time injection. Assumption 6 restricts the attacker to altering existing command packets. An attacker that can generate its own commands at arbitrary times can apply the continuous-time zero-dynamics input of an agent with $c_{1i}c_{2i} < 0$, whose zero $s = -c_{1i}/c_{2i}$ is unstable (Theorem 1(iv)). Such an attack keeps the output at zero at every instant, so no sampling-based defense, including [1], [8], and ARAS, can reveal it. The remedies lie elsewhere: authenticated command channels, traffic-based intrusion detection that flags packets the controller did not send [35]–[38], or an output with $c_{1i}c_{2i} > 0$ when the sensors allow it (Theorem 1(i)).

Sensor compromise. An attacker that also falsifies $\tilde{y}_i$ can mount a covert attack [4], which defeats any residual test. ARAS does not address this case.

Signal-to-noise ratio. The residual of a mismatched step is 1% to 2.3% of the deviation in our example (Section V-A). Performance therefore depends on the ratio between the damage threshold and the measurement noise. Table VI shows that ARAS, uniform randomization, and dual-rate sampling all lose protection as the noise grows, ARAS first.

Comparison with dual-rate sampling. Under Assumption 6, the dual-rate scheme [8] blocked every attack in our tests without extra messages. It needs a sensor sampled twice as fast as the actuator is updated, and a detector that processes intersample outputs. When the sensor rate is the binding constraint, which is common with cameras or satellite receivers, dual-rate sampling forces a slower control update; ARAS keeps sensing and actuation on the same clock and spends its overhead on messages instead. The two schemes can also be combined. We regard dual-rate sampling as a strong alternative in this threat model, and ARAS as the option when intersample sensing is not available.

Model scope. The analysis covers linear double-integrator agents with ZOH and an undirected graph. Nonlinear agents, directed or switching graphs, and attackers that combine ZDAs with packet loss remain open.

9. Conclusion

We have presented ARAS, an adaptive randomized asynchronous sampling mechanism with a distributed windowed χ^2 detector, for resilient consensus of heterogeneous MASs under schedule-aware ZDAs. We showed that the zeros of the sampled network split into single-agent zeros, that agents with $c_{1i}c_{2i} < 0$ have an unstable zero for every period, and that randomizing the period bounds the probability of a persistent perfectly stealthy attack by $(1 - (m - 1)\ p_min)^K$ for any attacker that does not know the current mode. The same argument gives a lower bound on the residual, and the guarantee disappears when the current mode leaks. Mean-square practical consensus holds for every admissible schedule under a windowed contraction condition. In 500-trial Monte Carlo tests on a four-quadrotor formation, ARAS reduced the success rate of a schedule-aware attacker from 99.8% under deterministic asynchronous sampling [1] to 5.6%, with 29.7% extra messages against 49.0%. Uniform randomization and dual-rate sampling [8] reached lower success rates, the

first at higher communication cost and the second at a doubled sensing rate. Performance degrades with measurement noise and with leaks of the current draw. Future work will address directed switching topologies, adversarial packet loss, nonlinear Euler–Lagrange agent dynamics, and combinations of ARAS with dual-rate sensing.

References

1. X. Liu, J. Xia, J. H. Park, and G. Chen, "Vulnerability analysis and asynchronous-based security countermeasure for sampled-data control multiagent systems under zero-dynamics attack," IEEE Trans. Ind. Informat., vol. 22, no. 4, pp. 2782–2792, Apr. 2026.
2. W. Yu, L. Zhou, X. Yu, J. Lü, and R. Lu, "Consensus in multi-agent systems with second-order dynamics and sampled data," IEEE Trans. Ind. Informat., vol. 9, no. 4, pp. 2137–2146, Nov. 2013.
3. W. He, W. Xu, X. Ge, Q.-L. Han, W. Du, and F. Qian, "Secure control of multiagent systems against malicious attacks: A brief survey," IEEE Trans. Ind. Informat., vol. 18, no. 6, pp. 3595–3608, Jun. 2022.
4. A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "A secure control framework for resource-limited adversaries," Automatica, vol. 51, pp. 135–148, Jan. 2015.
5. F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," IEEE Trans. Autom. Control, vol. 58, no. 11, pp. 2715–2729, Nov. 2013.
6. K. J. Åström, P. Hagander, and J. Sternby, "Zeros of sampled systems," Automatica, vol. 20, no. 1, pp. 31–38, 1984.
7. J. Kim, G. Park, H. Shim, and Y. Eun, "Zero-stealthy attack for sampled-data control systems: The case of faster actuation than sensing," in Proc. 55th IEEE CDC, Las Vegas, NV, USA, 2016, pp. 5956–5961.
8. M. Naghnaeian, N. H. Hirzallah, and P. G. Voulgaris, "Security via multirate control in cyber-physical systems," Syst. Control Lett., vol. 124, pp. 12–18, Feb. 2019.
9. J. Fu, G. Wen, W. Yu, T. Huang, and X. Yu, "Consensus of second-order multiagent systems with both velocity and input constraints," IEEE Trans. Ind. Electron., vol. 66, no. 10, pp. 7946–7955, Oct. 2019.
10. J. Kim, J. Back, G. Park, C. Lee, H. Shim, and P. G. Voulgaris, "Neutralizing zero dynamics attack on sampled-data systems via generalized holds," Automatica, vol. 113, 2020, Art. no. 108778.
11. G. Park, C. Lee, H. Shim, Y. Eun, and K. H. Johansson, "Stealthy adversaries against uncertain cyber-physical systems: Threat of robust zero-dynamics attack," IEEE Trans. Autom. Control, vol. 64, no. 12, pp. 4907–4919, Dec. 2019.
12. Y. Mao, H. Jafarnejadsani, P. Zhao, E. Akyol, and N. Hovakimyan, "Novel stealthy attack and defense strategies for networked control systems," IEEE Trans. Autom. Control, vol. 65, no. 9, pp. 3847–3862, Sep. 2020.
13. Y. Wang, J. Gao, Z. Zuo, Q. Han, and W. Zhang, "Periodic zero-dynamics attacks for discrete-time second-order multi-agent systems," Int. J. Robust Nonlinear Control, vol. 32, no. 9, pp. 5619–5636, 2022.
14. P. Griffioen, B. Krogh, and B. Sinopoli, "A moving target defense for securing cyber-physical systems," IEEE Trans. Autom. Control, vol. 66, no. 5, pp. 2016–2031, May 2021.
15. Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," IEEE Control Syst. Mag., vol. 35, no. 1, pp. 93–109, Feb. 2015.
16. T. Sui, Y. Mo, D. Marelli, X. Sun, and M. Fu, "The vulnerability of cyber-physical systems under stealthy attacks," IEEE Trans. Autom. Control, vol. 66, no. 2, pp. 637–650, Feb. 2021.
17. G. Chen, X. Liu, X. Ren, J. Xia, and J. H. Park, "Asynchronous-based countermeasure for stealthy attack on aperiodic sampled-data control systems," IEEE Trans. Autom. Control, vol. 70, no. 9, pp. 6159–6166, Sep. 2025.
18. R. Li, G. Ren, Q. Gan, J. Yang, Q. Kang, and J. Cao, "Event-based fixed-time secure control for multiagent systems under distributed DoS attacks," IEEE Trans. Ind. Informat., vol. 21, no. 4, pp. 2858–2867, Apr. 2025.
19. A. Nazarzadeh and H. J. Marquez, "Secure nonlinear sampled-data control system against stealthy attack: Multirate approach," IEEE Trans. Autom. Control, vol. 68, no. 12, pp. 8021–8028, Dec. 2023.
20. Z. Guo, D. Shi, K. H. Johansson, and L. Shi, "Worst-case stealthy innovation-based linear attack on remote state estimation," Automatica, vol. 89, pp. 117–124, 2018.
21. S. Weerakkody and B. Sinopoli, "Detecting integrity attacks on control systems using a moving target approach," in Proc. 54th IEEE CDC, Osaka, Japan, 2015, pp. 5820–5826.
22. H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," IEEE J. Sel. Areas Commun., vol. 31, no. 4, pp. 766–781, Apr. 2013.
23. Y. Chen, S. Kar, and J. M. F. Moura, "Resilient distributed estimation: Sensor attacks," IEEE Trans. Autom. Control, vol. 64, no. 9, pp. 3772–3779, Sep. 2019.
24. W. Ao, Y. Song, and C. Wen, "Distributed secure state estimation and control for CPSs under sensor attacks," IEEE Trans. Cybern., vol. 50, no. 1, pp. 259–269, Jan. 2020.

25. R. Zhao, Z. Zuo, Y. Shi, Y. Wang, and W. Zhang, "DoS and stealthy deception attacks for switched systems: A cooperative approach," IEEE Trans. Autom. Control, vol. 69, no. 7, pp. 4396–4410, Jul. 2024.
26. K. Kimura and H. Ishii, "Quantized zero dynamics attacks against sampled-data control systems," IEEE Trans. Autom. Control, vol. 69, no. 5, pp. 3418–3425, May 2024.
27. X. Xu, Y. Zhu, F. Liao, and X. Zhao, "Sampled-data control of switched affine systems with applications to DC–DC converters," IEEE Trans. Ind. Informat., vol. 21, no. 3, pp. 2442–2450, Mar. 2025.
28. Y. Lei, Z. Wang, X. Wang, H. Li, and I.-C. Morărescu, "Hybrid formation control for heterogeneous uncertain linear two-time-scale systems," Automatica, vol. 176, 2025, Art. no. 112267.
29. Z. Wang, H. Zhang, X. Cao, E. Liu, H. Li, and J. Zhang, "Modeling and detection scheme for zero-dynamics attack on wind power system," IEEE Trans. Smart Grid, vol. 15, no. 1, pp. 934–943, Jan. 2024.
30. X.-X. Ren, G.-H. Yang, and X.-G. Zhang, "Optimal stealthy attack with historical data on cyber-physical systems," Automatica, vol. 151, 2023, Art. no. 110895.
31. M. A. Dahleh and I. J. Diaz-Bobillo, Control of Uncertain Systems: A Linear Programming Approach. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.
32. A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, "Revealing stealthy attacks in control systems," in Proc. 50th Allerton Conf. Commun., Control, Comput., Monticello, IL, USA, 2012, pp. 1806–1813.
33. E. Davison and S. Wang, "Properties and calculation of transmission zeros of linear multivariable systems," Automatica, vol. 10, no. 6, pp. 643–658, 1974.
34. E. B. Wilson, "Probable inference, the law of succession, and statistical inference," J. Amer. Statist. Assoc., vol. 22, no. 158, pp. 209–212, 1927.
35. I. Mahmood et al., "Intrusion detection in 5G cellular network using machine learning," Comput. Syst. Sci. Eng., vol. 47, no. 2, pp. 2439–2453, 2023, doi: 10.32604/csse.2023.033842.
36. A. Amjad et al., "Detection and mitigation of DDoS attack in cloud computing using machine learning algorithm," EAI Endorsed Trans. Scalable Inf. Syst., vol. 6, no. 23, 2019, doi: 10.4108/eai.29-7-2019.159834.
37. K. Alissa et al., "Botnet attack detection in IoT using machine learning," Comput. Intell. Neurosci., vol. 2022, Art. no. 4515642, 2022, doi: 10.1155/2022/4515642.
38. S. Y. Ali et al., "Securing cloud environments: A convolutional neural network (CNN) approach to intrusion detection system," J. Comput. Biomed. Informat., vol. 6, no. 2, pp. 295–308, 2024, doi: 10.56979/602/2024.