

Semantic Drift in AI-Native 6G Multi-Agent Systems: A Systematic Review and Taxonomy of Mitigation Strategies

Odesa Ogaga Edward¹, Afolabi Awodeyi², Ugbeh Raymond Nduka³, Omolegho Ibok⁴, Idama Omokaro⁵

^{1,2,3,4,5}Department of Computer Engineering, Southern Delta University Ozoro, Delta State Nigeria, odesaedwardogaga@gmail.com, <https://orcid.org/0009-0006-8259-9812>

Abstract

AI-native sixth-generation (6G) wireless networks are envisioned to orchestrate massive populations of autonomous agents—including unmanned aerial vehicles (UAVs), connected autonomous vehicles (CAVs), industrial robotics, and distributed Internet of Things (IoT) nodes—that transcend traditional bit-pipe transmissions to exchange rich, task-oriented semantic representations. Extracted via advanced edge artificial intelligence (AI), transmitted across learned air interfaces, fused into shared knowledge bases, and ultimately translated into collaborative joint actions, this semantic processing pipeline is inherently vulnerable to compounding imperfections. Sensors degrade through calibration loss and environmental wear, edge inference models suffer from severe distribution shifts, semantic encoders and decoders experience mutual misalignment, wireless channels distort latent feature spaces, knowledge bases diverge across heterogeneous vendors, and malicious or faulty Byzantine peers inject persistent false beliefs. We formally define the resulting gradual degradation of alignment between representation and reality, as well as inter-agent consensus divergence, as semantic drift. This comprehensive review traces the entire sensing-to-decision pipeline across seven distinct stages ($S_1 \rightarrow S_7$). We introduce a rigorous stage-wise formalization accompanied by an end-to-end analytical error bound demonstrating how localized errors propagate and amplify downstream. Furthermore, we present a structured *detect–prevent–correct–contain* taxonomy of mitigation mechanisms for each stage, evaluate numerical models contrasting vulnerable linear consensus with Byzantine-resilient fusion, outline a closed-loop cross-stage management architecture with joint resource allocation formulations, consolidate evaluation metrics, and articulate open research challenges.

Index Terms: *Semantic drift, AI-native 6G, semantic communication, edge AI, multi-agent systems, federated learning, Byzantine-resilient consensus, age of incorrect information.*

I. Introduction

Sixth-generation (6G) wireless systems are rapidly transitioning from conceptual frameworks to concrete architectural blueprints characterized by pervasive machine learning embedded directly into the physical layer, protocol stacks, and network management engines [1]. Rather than treating the wireless channel as a static bit pipe designed solely for lossless symbol reproduction, AI-native 6G paradigms enable transceivers, resource allocation schedulers, and application layers to learn and adapt jointly [1], [2]. This evolution aligns seamlessly with the emergence of semantic and task-oriented communications, where the primary objective is to convey information that is strictly relevant to downstream multi-agent tasks rather than replicating raw input bitstreams [2].

When the endpoints of these semantic networks consist of collaborative teams of autonomous agents, the operational benefits are profound. Decentralized agents sense their immediate surroundings, extract compact semantic features at the network edge, exchange these abstractions over low-latency 6G links, and arrive at unified operational decisions with substantially lower communication overhead than raw data sharing would ever permit. However, this tightly coupled architecture introduces a critical and hitherto under-explored reliability challenge: **semantic drift**.

In conventional communication architectures, transmission errors are treated as independent, memoryless random variables mitigated via channel coding and automatic repeat requests (ARQ). In contrast, semantic multi-agent systems operate via cascading sequences of learned neural approximations. Consequently, errors within semantic pipelines are highly structured, persistent, and cumulative. A miscalibrated sensor repeatedly injects biased environmental features; a stale codebook continuously misinterprets semantic tokens; and a faulty or adversarial peer consistently broadcasts misleading beliefs. Because agents continuously fuse, update, and re-share information across the network, minor initial discrepancies compound rapidly, leading to network-wide consensus failure.

Although isolated facets of this phenomenon—such as concept drift, domain adaptation, and knowledge base heterogeneity—have been studied extensively within machine learning and distributed systems, treating semantic drift as a unified, end-to-end systemic property across the entire sensing-to-decision pipeline remains absent from current literature.

This paper contributions include:

1. We establish a unified, seven-stage formalization of the multi-agent semantic pipeline (S_1 to S_7) and derive a novel end-to-end analytical error bound governed by stage-wise Lipschitz constants.
2. We propose a comprehensive *detect–prevent–correct–contain* taxonomy classifying existing mitigation strategies across all processing tiers.
3. We provide detailed numerical evaluations contrasting vulnerable linear consensus against robust, Byzantine-resilient multi-agent fusion rules.
3. We formulate a closed-loop cross-stage drift management architecture coupled with a joint compute-communication resource allocation optimization problem.
4. We consolidate performance evaluation metrics, identify current benchmark limitations, and chart key open research avenues for dependable AI-native 6G semantic networks.

1.

II. Background and Related Surveys

A. AI-Native 6G Architectures

AI-native 6G embeds machine learning natively across all protocol layers, leveraging distributed edge computing infrastructure to execute complex inference tasks in close proximity to data sources [1]. Key enablers include Integrated Sensing and Communication (ISAC), which dual-purposes radio frequency waveforms to simultaneously provide high-throughput communication and high-resolution environmental sensing and radar-based positioning references [7].

Furthermore, network digital twins create synchronized, real-time virtual replicas of physical network topologies and propagation environments, enabling predictive resource orchestration and proactive fault management [6].

B. Semantic Communications and Multi-Agent Learning

Deep joint source-channel semantic coding transceivers have demonstrated exceptional compression efficiency for text, speech, and visual modalities by leveraging neural network approximations optimized via information bottleneck principles [2], [8], [12]. In parallel, multi-agent reinforcement learning (MARL) and collaborative perception frameworks enable distributed agents (such as connected autonomous vehicles and drone swarms) to exchange intermediate feature maps, semantic landmarks, and spatial occupancy grids to execute complex coordinated maneuvers [13].

C. Positioning Against Existing Reviews

While extensive surveys exist on edge artificial intelligence [1], semantic communication foundations [2], wireless federated learning [5], and collaborative perception [13], none synthesize these domains through the unifying lens of end-to-end semantic drift. This review bridges that critical gap.

III. The Core Chain and a Stage-Wise Drift Model

A. The Seven-Stage Chain

The end-to-end semantic pipeline in multi-agent AI-native 6G networks is modeled as a sequential composition of seven distinct processing stages:

Fig. 1 shows the chain studied here. Let w denote the true world state. Stage S_k implements a map f_k , so the ideal end-to-end pipeline is

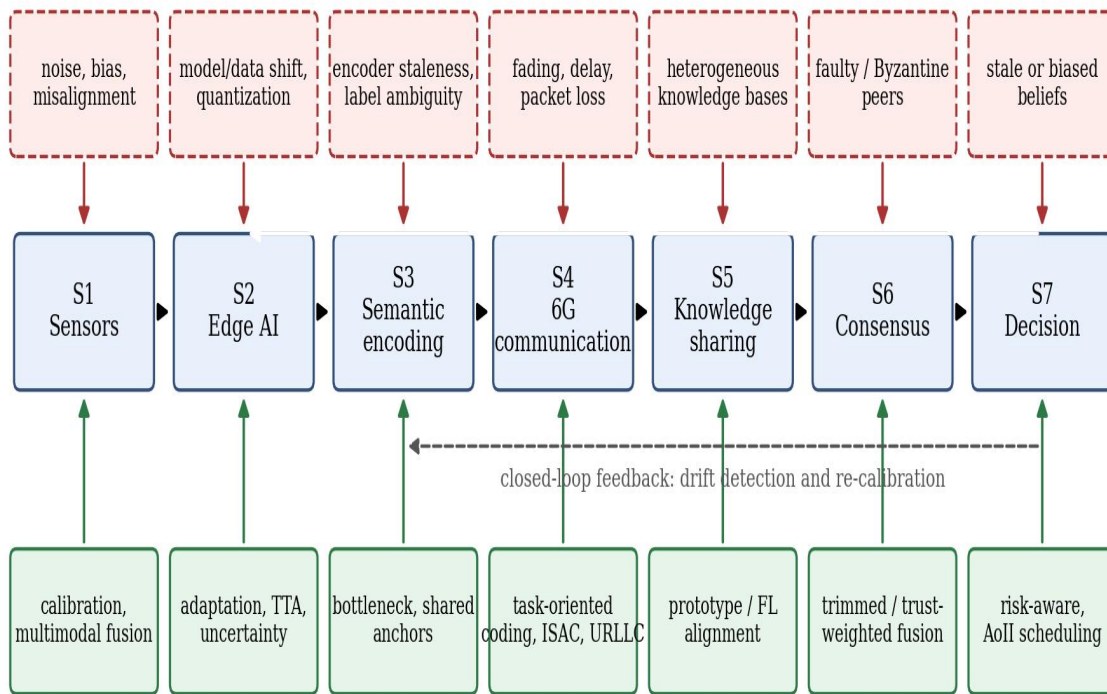
$$u_7 = f_7 \circ f_6 \circ \dots \circ f_1(w), \quad u_k = f_k(u_{k-1}), \quad u_0 = w, \quad (1)$$

where u_1 is the sensor measurement, u_2 the edge-AI inference, u_3 the semantic code, u_4 the received code, u_5 the shared knowledge, u_6 the consensus belief, and u_7 the decision. In a real system each stage is realized by an approximator $\tilde{f}_k$ whose deviation from the intended behaviour at time t is

$$\sup_u \|\tilde{f}_k(u;t) - f_k(u)\| \leq \varepsilon_k(t), \quad (2)$$

and $\varepsilon_k(t)$ typically *grows* over time without intervention. This is the formal sense in which drift occurs: $\varepsilon_k(t)$ is not a fixed quantization or noise floor, but a trend driven by environmental change, model ageing, or adversarial influence.

Drift sources



Mitigation families

Fig. 1. The core chain from sensors to decision, with the main drift sources entering each stage (top) and the principal mitigation families (bottom). A feedback path supports detection and re-calibration.

S₁ (Sensors): Raw physical data acquisition, analog-to-digital conversion, and multi-modal alignment.

S₂(Edge AI): Local feature extraction, object detection, and edge inference.

S₃ (Semantic Encoding): Task-oriented compression, symbol mapping, and codebook translation.

S₄ (6G Communication): Air interface transmission, ISAC-assisted propagation, and packet scheduling.

S₅ (Knowledge Sharing): Decentralized peer-to-peer prototype aggregation and federated alignment.

S₆ (Consensus): Byzantine-resilient belief pooling and distributed agreement protocols.

S₇ (Decision): Risk-aware action selection and digital twin rehearsal.

B. End-to-End Error Bound

To characterize how local stage imperfections propagate across the network, we establish the following analytical result.

Assume each ideal map f_k is L_k -Lipschitz, i.e., it amplifies input perturbations by at most L_k (with $L_k < 1$ for contracting operations such as averaging or verified fusion, and $L_k > 1$ for operations that magnify perturbations). Then we have the following result.

Proposition 1. Under the stated assumptions, the end-to-end deviation satisfies

$$\| \tilde{u}_7 - u_7 \| \leq \sum_{k=1}^7 \left(\prod_{j=k+1}^7 L_j \right) \varepsilon_k(t). \quad (3)$$

Proof sketch. Define $e_k = \| \tilde{u}_k - u_k \|$ with $\tilde{u}_k = \tilde{f}_k(\tilde{u}_{k-1}; t)$. By the triangle inequality, $e_k \leq \| \tilde{f}_k(\tilde{u}_{k-1}) - f_k(\tilde{u}_{k-1}) \| + \| f_k(\tilde{u}_{k-1}) - f_k(u_{k-1}) \| \leq \varepsilon_k + L_k e_{k-1}$ with $e_0 = 0$. Unrolling the recursion gives (3). $\square$

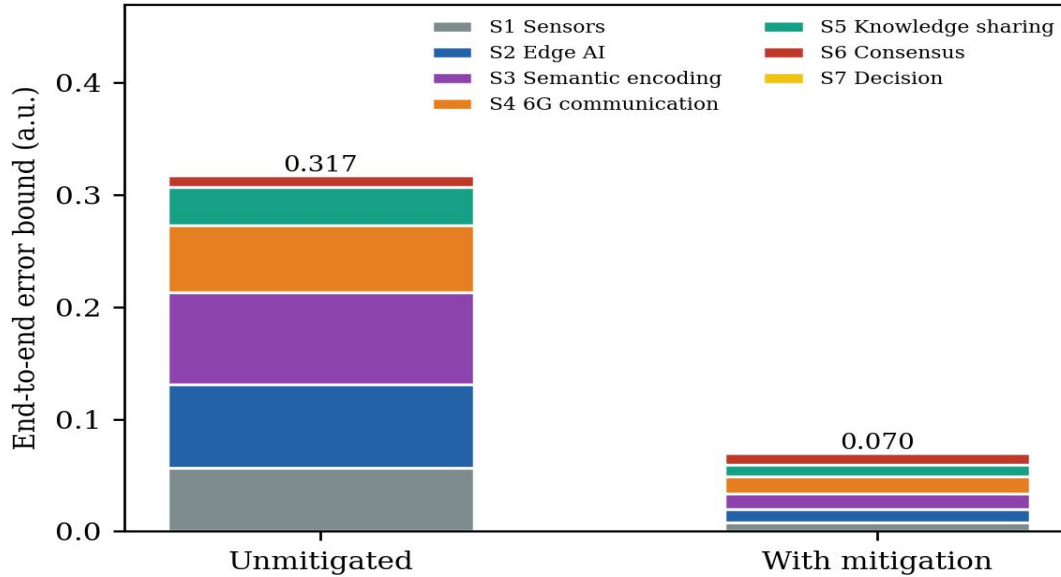


Fig. 2. Illustrative end-to-end error budget from (3). Bar segments show each stage's contribution $\varepsilon_k \prod_{j>k} L_j$. Parameter values are illustrative, chosen by the authors.

This analytical bound carries two vital implications:

1. Errors injected in early stages (S_1 through S_3) are severely amplified if downstream Lipschitz constants exceed unity ($L_j > 1$). so *early* errors are the most dangerous unless later stages contract.
2. Network resilience requires incorporating contraction layers ($L_j < 1$)—such as robust consensus and uncertainty-gated fusion—at later stages to actively shrink error propagation bounds.

Fig. 2 evaluates (3) with illustrative values: with no mitigation, errors injected at the semantic encoder and edge AI dominate the total; with mitigation (which both reduces ε_k and brings L_k of the knowledge-sharing and consensus stages below one) the total shrinks by roughly a factor of four. The values are chosen for illustration and are not measurements.

C. Defining Semantic Drift

We formalize semantic drift across two complementary dimensions: We distinguish two forms of drift. *Temporal drift* of agent i is the divergence between the current and the reference joint distribution of its inputs and labels,

$$\delta_t^i = D(P_t^i(x, y) \| P_0^i(x, y)), \quad (4)$$

where D is a divergence (for example KL, Wasserstein, or maximum mean discrepancy). *Inter-agent drift* is the disagreement between two agents' semantic interpretation of the same input,

$$\Delta_{ij}(t) = E_x \left[d_s \left(g_i(f_i(x)), g_j(f_j(x)) \right) \right], \quad (5)$$

with encoder f_i , task head g_i , and semantic distance d_s . Both are needed: an isolated agent can drift temporally without any inter-agent signal, while a fleet whose agents drift in *different* directions shows rising Δ_{ij} even if each stays internally consistent. Concept-drift research provides detection and adaptation tools for the first form [13], [14]; the second form is specific to semantic multi-agent networks and is the main subject of Sections IV-E and IV-F.

Temporal Drift: Quantified via statistical divergence (such as Maximum Mean Discrepancy or Kullback-Leibler divergence) between an agent's current input-label distribution $P_t^i(x,y)$ and its baseline distribution $P_0^i(x,y)$ [10]:

$$\delta_t^i = D(P_t^i(x,y) \parallel P_0^i(x,y)), \quad (4)$$

Inter-Agent Drift: Measures semantic inconsistency between distinct peer agents i and j processing identical observations:

$$\Delta_{ij}(t) = E_x \left[d_s \left(g_i(f_i(x)), g_j(f_j(x)) \right) \right], \quad (5)$$

with encoder f_i , task head g_i , and semantic distance d_s . Both are needed: an isolated agent can drift temporally without any inter-agent signal, while a fleet whose agents drift in *different* directions shows rising Δ_{ij} even if each stays internally consistent

IV. Stage-Wise Review of Drift and Mitigation

A. S₁ (Sensors)

Sensor calibration loss, thermal aging, clock skew, and inter-agent pose misalignment which severely degrade collaborative perception frameworks [13]. Mitigation include robust multi-modal sensor fusion and ISAC-enabled radio positioning reference anchors that dynamically correct sensor coordinate frames [7].

B. S₂ (Edge AI)

Edge inference models encounter severe distribution shifts between training distributions and deployment environments [1], [11]. Statistical divergence is monitored continuously using Kernel Two-Sample Tests via Maximum Mean Discrepancy (MMD) [10]:

$$\text{MMD}^2(P, Q) = E k(z, z') + E k(w, w') - 2 E k(z, w), \quad (6)$$

with $z, z' \sim P$, $w, w' \sim Q$, and kernel k . Predictive uncertainty is quantified using Monte Carlo dropout [9] and deep ensembles, while test-time adaptation (TTA) via entropy minimization corrects online model drift [11].

C. S₃ (Semantic Encoding)

Semantic transceivers optimize information bottleneck trade-offs to retain task-critical features while discarding redundant bitstreams [12]. The trade-off between compression and fidelity can be expressed as a semantic rate-distortion function,

$$R(D) = \min_{p(z|x): E[d_s(x, \hat{x})] \leq D} I(X; Z), \quad (7)$$

where d_s measures semantic rather than bitwise distortion [12],

Standardized open-vocabulary foundation embedding spaces mitigate encoder-decoder semantic mismatch across heterogeneous hardware vendors.

D. S₄ (6G Communication)

Channel impairments, delay, and packet loss distort latent features. With a received signal $y = \sqrt{\gamma} h z + n$, noise $n \sim \text{CN}(0, \sigma^2 I)$, and fading gain h , the achievable payload per update is limited by capacity,

$$b_i \leq T_c B \log_2 (1 + \text{SNR}_i), \quad \text{SNR}_i = \frac{\gamma |h_i|^2 P_i}{\sigma^2}, \quad (8)$$

so poor channels force coarser features, trading directly against semantic fidelity. Delay creates *staleness*, which is harmful only if the information has become wrong. The age of incorrect information captures this by weighting elapsed time with an error indicator [3]:

$$\Delta_{\Pi}(t) = f(t - V_t) g(s_t, \hat{s}_t), \quad (9)$$

where V_t is the last instant at which the estimate $\hat{s}$ matched the source s , f increases with elapsed time, and g is a mismatch penalty. Conventional age of information metrics fail to capture semantic validity; hence, the Age of Incorrect Information (AoII) is employed [3].

E. S₅ (Knowledge Sharing)

Federated learning across wireless multi-agent networks suffers from non-identically distributed (Non-IID) local data [5]. Agents mitigate this by exchanging compact class prototypes μ_c^i and penalize disagreement, and regularizing local training objectives toward shared global representations [5]:

$$\min_{\{\theta_i\}} \sum_i p_i F_i(\theta_i) + \lambda \sum_i \sum_c \|\mu_c^i - \mu_c\|^2, \quad \mu_c = \sum_i p_i \mu_c^i, \quad (10)$$

where F_i is the local loss and p_i the aggregation weight

F. S₆ (Consensus)

Consensus is the stage where local drift becomes network-wide drift. Standard linear consensus drives all agents to a weighted average,

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j \in \mathcal{N}_i} a_{ij} (x_j(k) - x_i(k)), \quad (11)$$

which converges at a rate governed by the second-largest eigenvalue modulus of the weight matrix [4]. The same property makes it fragile: a single agent that persistently broadcasts a wrong value pulls *every* honest agent toward it. In semantic settings the “value” may be a class posterior or an embedding rather than a scalar, and a faulty peer can be internally consistent yet wrong.

Resilient consensus discards extreme neighbour values before averaging. In the W-MSR rule [15], agent i removes up to F neighbour values larger than its own and up to F smaller ones, and averages the rest:

$$x_i(k+1) = \sum_{j \in (N_i \cup \{i\}) \setminus \mathcal{R}_i(k)} w_{ij}(k) x_j(k), \quad (12)$$

where $\mathcal{R}_i(k)$ is the removed set. If the network is sufficiently robust, honest agents converge to a value inside the range of honest initial states despite up to F adversaries, a guarantee unavailable to plain averaging. For distributions, a *log-opinion pool* fuses neighbour beliefs multiplicatively with weights that can encode trust,

$$p_{\text{fused}}(c) \propto \prod_i p_i(c)^{w_i}, \quad \sum_i w_i = 1, \quad (13)$$

and when cross-correlations between estimates are unknown, covariance intersection avoids double counting :

$$P^{-1} = \omega P_a^{-1} + (1-\omega) P_b^{-1}, \quad \omega \in [0, 1]. \quad (14)$$

Byzantine-tolerant aggregation rules from distributed learning provide further defenses [14], and ledger-based mechanisms offer decentralized trust at the cost of latency and energy [15].

Fig. 4 shows a 12-agent circulant network (each agent linked to the three nearest neighbours on each side) in which one agent persistently broadcasts the value 5 while honest agents start in $[0.09, 0.80]$ (single run, fixed random seed). Under plain averaging, all honest agents are dragged to approximately 4.94, essentially the adversary's value. With W-MSR ($F=1$), honest agents converge within a few iterations to 0.45, inside the honest initial range. This is a deliberately simple scalar example, not a benchmark, but it demonstrates the contrast formalized by (11)–(12).

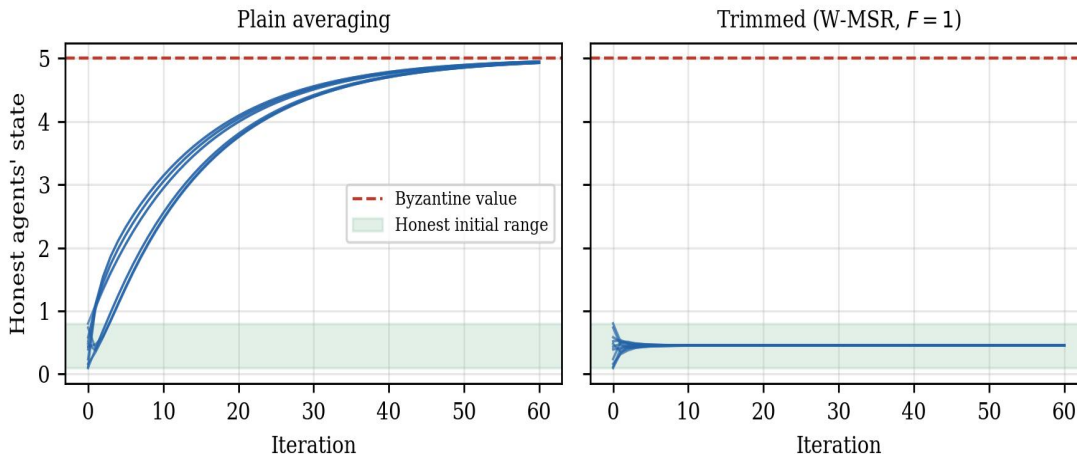


Fig. 4. Consensus with one Byzantine agent (value 5). Left: plain averaging is captured by the adversary. Right: trimmed W-MSR fusion ($F=1$) keeps honest agents within their initial range.

G. S₇ (Decision)

Downstream actions inherit accumulated upstream drift. Total variation distance bounds guarantee that decision regret scales linearly with belief error, motivating risk-aware fallback policies and digital twin simulation rehearsals [6].

Table II. Stage-wise summary of drift sources and mitigation (representative works).

Stage	Main drift sources	Representative mitigation
-------	--------------------	---------------------------

Stage	Main drift sources	Representative mitigation
S1 Sensors	Calibration, pose/clock misalignment, modality gap	Multimodal fusion ; ISAC reference ; pose-aware alignment
S2 Edge AI	Distribution shift, compression, split staleness	MMD and uncertainty monitoring ; TTA ; continual learning
S3 Semantic encoding	Information loss, encoder–decoder mismatch	Bottleneck, ; shared foundation embeddings ; KB sync
S4 6G comm.	Fading, delay, loss, staleness	AI-native air interface ; AoII scheduling , task-oriented coding
S5 Knowledge sharing	Non-IID data, protocol drift, label mismatch	Wireless F ; selective sharing
S6 Consensus	Faulty peers, double counting	W-MSR; covariance intersection; Byzantine-robust rules
S7 Decision	Biased or stale beliefs	Risk-aware rules; digital-twin rehearsal ; robust MARL

V. Cross-Stage Drift Management & Closed-Loop Architecture

Stage-wise interventions alone are insufficient to guarantee network dependability. We propose a closed-loop approach.

A. Closed-Loop Architecture

A closed-loop architecture featuring continuous *detect* $\rightarrow$ *diagnose* $\rightarrow$ *correct* loops coordinated via an edge-hosted knowledge plane and digital twin infrastructure [6]. Fig. 5 organizes drift management as a loop of *detect* $\rightarrow$ *diagnose* $\rightarrow$ *correct*. Detection uses the stage-level monitors described above (MMD on latent features, uncertainty, AoII, inter-agent disagreement Δ_{ij}). Diagnosis localizes the alarm to a stage and source, for example by comparing an agent's feature statistics with those of the digital twin, or its beliefs with its peers'. Correction is then targeted: adaptation of an edge model (S2), re-synchronization of an encoder–decoder pair (S3), re-weighting or exclusion of a peer (S6), or re-anchoring using ISAC measurements (S1). An edge-hosted knowledge plane holds reference models, codebooks, and trust scores so that these corrections can be coordinated without a central controller on the critical path.

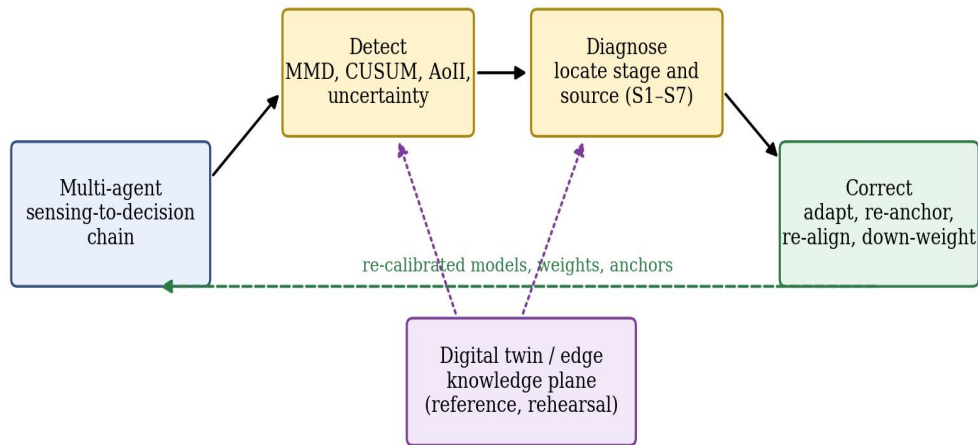


Fig. 5. Closed-loop drift management across the chain, supported by a digital twin and edge knowledge plane.

B. Joint Allocation

Furthermore, resource allocation is formulated as a joint optimization problem balancing communication rates r_k , compute power f_k , and transmission power p_k :

$$\min_{\{r_k, f_k, p_k\}} \sum_{k=1}^7 \omega_k \varepsilon_k(r_k, f_k, p_k) \quad \text{s.t.} \quad \sum_k r_k \leq C, \quad \sum_k \tau_k \leq T_{\max}, \quad \sum_k E_k \leq E_{\max}, \quad (16)$$

where weights E_k reflect downstream Lipschitz error amplification factors.

VI. Evaluation Metrics and Benchmarks

Table III proposes metrics for each stage together with end-to-end indicators. Rigorous validation requires standardized multi-tier evaluation metrics spanning sensor calibration error, MMD feature divergence [10], semantic distortion d_s , Age of Incorrect Information [3], inter-agent drift Δ_{ij} , W-MSR resilience bounds [4], and decision regret. Comprehensive testbeds integrating hardware-in-the-loop network emulation are essential to benchmark end-to-end semantic drift dynamics.

Table III. Proposed metrics along the chain.

Stage	Metric	What it captures
S1	Calibration error, pose/clock error	Sensor and alignment drift
S2	MMD, expected calibration error, accuracy under shift	Model drift
S3	Semantic distortion d_s , bits per task success	Encoder fidelity, efficiency
S4	AoII, task-weighted latency, feature error rate	Freshness and correctness
S5	Inter-agent drift Δ_{ij} , prototype distance	Knowledge alignment

Stage	Metric	What it captures
S6	Honest-agent disagreement, deviation from honest range	Resilience to faulty peers
S7	Regret, safety violations, time to detect drift	Decision quality
End to end	Error-bound tightness vs. (3), drift-detection delay	System-level behaviour

VII. CHALLENGES AND FUTURE DIRECTIONS

- Theory of End-to-End Drift:** Current proofs (Proposition 1) are limited to predictable, fixed stages. Scaling this to unpredictable, learned, or constantly changing environments while determining exact boundaries L_k for neural networks is still an open problem.
- Drift Diagnosis:** Spotting system drift is straightforward, but pinpointing which stage caused it is difficult. Future systems require methods like latent-space checks, peer comparisons, and digital twins to locate the root cause without relying on manual ground-truth labels
- Knowledge-Base Consistency at Scale:** Keeping background data synchronized across different multi-vendor agents without slowing down the network is an active challenge [11], [16]. Shared embedding models ease this but add bottlenecks in size, speed, and overall security trust [54].
- Semantic-Level Security:** Traditional defenses handle basic data manipulation well [42], [64]. However, smart, realistic adversarial messages that spoof meaning while evading standard detection filters require entirely new defense frameworks, potentially backed by ledger-based security tracking [65].
- Energy and Sustainability:** Tracking and correcting systems at the network edge drains battery power. Resource equations—like equation (16)—must account for energy costs directly so optimization can run natively on low-power devices.
- Safety-Critical Decisions:** Real-world autonomous vehicles and drone swarms cannot rely on warnings alone. Drift detections must instantly trigger mathematically secure fallback operations—such as those in equation (15)—to guarantee immediate safety.

VII. Conclusion

Semantic drift represents an inherent systemic challenge across the entire sensing-to-decision chain in AI-native 6G multi-agent networks. By establishing rigorous stage-wise error bounds, deploying Byzantine-resilient consensus protocols [4], and implementing closed-loop cross-stage coordination architectures [6], distributed autonomous systems can achieve unprecedented levels of operational reliability and semantic fidelity.

References

- [1] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6G: Vision, enabling technologies, and applications," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 5–36, 2022. <https://doi.org/10.1109/jsac.2021.3126076>

- [2] Yang, W., Du, H., Liew, Z. Q., Lim, W. Y. B., Xiong, Z., Niyato, D., Chi, X., Shen, X. S., & Miao, C. (2023). Semantic communications for future internet: Fundamentals, applications, and challenges. *IEEE Communications Surveys & Tutorials*, 25(1), 213–250. <https://doi.org/10.1109/COMST.2022.3223224>
- [3] A. Maatouk, S. Kriouile, M. Assaad, and A. Ephremides, "The age of incorrect information: A new performance metric for status updates," *IEEE/ACM Transactions on Networking*, vol. 28, no. 5, pp. 2215–2228, Oct. 2020, doi: 10.1109/TNET.2020.3005549
- [4] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013. <https://doi.org/10.1109/jsac.2013.130413>
- [5] P. Kairouz *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021. <https://doi.org/10.1561/22000000083>.
- [6] L. U. Khan, Z. Han, W. Saad, E. Hossain, M. Guizani, and C. S. Hong, "Digital twin of wireless systems: Overview, taxonomy, challenges, and opportunities," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2230–2254, 2022. <https://doi.org/10.1109/comst.2022.3198273>
- [7] Liu, F., Cui, Y., Masouros, C., Xu, J., Han, T. X., Eldar, Y. C., & Buzzi, S. (2022). Integrated sensing and communications: Toward dual-functional wireless networks for 6G and beyond. *IEEE Journal on Selected Areas in Communications*, 40(6), 1728–1767. <https://doi.org/10.1109/JSAC.2022.3156632>
- [8] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021. <https://doi.org/10.1109/TSP.2021.3071210>
- [9] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. International Conference on Machine Learning (ICML)*, 2016, pp. 1050–1059. <https://arxiv.org/abs/1506.02142>
- [10] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. 25, pp. 723–773, 2012. <https://jmlr.org/papers/v13/gretton12a.html>
- [11] Wang, D., Shelhamer, E., Liu, S., Olshausen, B., & Darrell, T. (2021). Tent: Fully test-time adaptation by entropy minimization. *International Conference on Learning Representations (ICLR)*. <https://openreview.net/forum?id=uXl3bZLkr3c>

- [12] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep variational information bottleneck," in *Proc. International Conference on Learning Representations (ICLR)*, 2017. <https://arxiv.org/abs/1612.00410>
- [13] R. Xu, H. Xiang, X. Tu, X. Xia, M.-H. Yang, and J. Ma, "V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer," in *Proc. European Conference on Computer Vision (ECCV)*, 2022, pp. 107–124. https://doi.org/10.1007/978-3-031-19842-7_7
- [14] Blanchard, P., El Mhamdi, E. M., Guerraoui, R., & Stainer, J. (2017). Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in Neural Information Processing Systems*, 30, 119–129. <https://proceedings.neurips.cc/paper/2017/hash/f4b9ec30ad9f68f89b29639786cb62ef-Abstract.html>
- [15] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proc. American Control Conference*, 1997, pp. 2369–2373. <https://ieeexplore.ieee.org/document/609105>