

Towards Autonomous Resilience: Architecting Agentic Middleware for the Cloud-Edge Continuum

Lovro Posarić, Tara Nikolić
University of Zagreb
Faculty of Organization and Informatics
IoT Laboratory
Pavlinska 2, 42000 Varaždin
{lposaric, tnikolic}@foi.hr

Abstract. *The Cloud-Edge-IoT continuum reduces latency for smart-city services but also introduces operational fragility: cascading, cross-tier failures can disrupt orchestration frameworks designed for steady-state optimisation. This paper introduces a three-tier agentic middleware coupling lightweight deterministic edge heuristics with selective escalation to a Cloud-tier orchestrator. A deterministic Escalation Score, constrained-decoding output schema, pre-execution validation, and decision logging bound the escalation pathway and make it reconstructible after an event. Alongside the architectural specification, we contribute a failure-first evaluation paradigm designed to quantify the cost of failure against the cost of intelligence through discrete-event simulation over open-data-derived urban dependency graphs. The paper is positioned as a design contribution: the architecture and evaluation framework are specified at the design level, while the empirical study is planned rather than completed.*

Keywords. Cloud-Edge continuum, smart-city infrastructure, agentic middleware, fault tolerance, large language models

1 Introduction

Modern smart cities increasingly distribute computation across the Cloud-Edge-IoT (CEI) continuum rather than concentrating it centrally, because the growing volume of devices and cognitive-service demand has outgrown what a centralised Cloud can absorb on its own (Chaturvedi et al., 2026; Wei et al., 2026). Pushing computation to the network edge reduces latency and bandwidth bottlenecks, enabling real-time urban services. However, distributing computing power across multi-operator, heterogeneous environments sharply increases operational vulnerability, and Cloud-oriented orchestration tools were not designed for the extreme device heterogeneity and dynamic conditions of the continuum (Belcastro et al., 2026). Current orchestration literature addresses dynamic placement, monitoring, and

adaptation, but evaluations provide limited evidence on containment of correlated compound failures propagating across service dependencies and infrastructure tiers (Angelis & Kousiouris, 2025), leaving an open question about operational resilience under compound, cross-tier failures.

Consider a storm that unfolds progressively. Rising water causes some Far-Edge acoustic sensors to stop reporting while others continue to transmit plausible but corrupted baseline values. Because conventional gateways act as passive forwarders, missing and integrity-violating readings can propagate upward without coordinated handling. As the storm severs a fibre link, the Smart-Edge tier is partitioned from the Cloud, the static orchestrator becomes unreachable, and the unhandled anomalies cascade, exhausting local resources and driving critical emergency-routing services into severe latency spikes. This architectural fault stems from an inability to reason about collapsing cross-tier dependencies. Beyond single-node recovery, such systems would benefit from a layer that recognises compound failures and coordinates reconfiguration before the cascade propagates widely.

The proposed architecture keeps its escalation and governance mechanisms separate from the domain models, so the former transfer across Cloud-Edge-IoT domains while the dependency, criticality, workload, and fault models remain specific to the smart-city setting. Two smart-city properties drive the design directly. First, multi-operator custody means no individual operator holds the complete service-dependency graph. The design therefore assumes a municipality-level platform that composes a versioned structural graph from operator declarations. The structural graph is shared, whereas live telemetry stays local to each Smart-Edge district. Each escalation therefore carries the relevant topology and criticality context. Second, physical-world events such as storms and accidents can produce correlated, potentially out-of-distribution failures across independently managed services, motivating the novelty-aware trigger and failure-first evaluation. Mixed criticality constrains the permitted recovery actions, while heterogeneity and scale shape implementation require-

ments but are not unique to smart cities. Addressing these limitations calls for an orchestration layer that can reason about collapsing cross-tier dependencies and coordinate reconfiguration beyond single-node recovery. This paper makes three contributions:

1. Architectural design: The formal specification of a three-tier agentic middleware with a transparent, threshold-governed escalation model for safely integrating Large Language Model (LLM) reasoning into physical infrastructure.
2. Theoretical foundation: A formal escalation criterion consisting of a deterministic score over a compact, interpretable system-state vector that defines the boundary at which reactive edge heuristics escalate to Cloud-tier semantic reasoning.
3. Methodological framework: A failure-first evaluation paradigm that quantifies the “cost of intelligence” against the “cost of failure.”

This is a design and blueprint contribution. The evaluation framework specified below has not yet been executed, so the hypotheses are stated as planned tests, and the paper reports no empirical results.

2 Background and related work

Literature predominantly models the CEI continuum as a hierarchy reflecting differences in hardware and network reliability (Gkonis et al., 2026). At the perimeter, the Far-Edge comprises massive deployments of constrained sensors and microcontrollers; relying purely on centralised Cloud processing introduces communication overhead and single points of failure. A Smart-Edge tier of robust localised nodes (e.g., lamppost-mounted edge nodes) therefore bridges the fragile Far-Edge and the distant Cloud, and the consensus is that it must possess high local autonomy to execute latency-constrained workloads without continuous Cloud validation (Liu et al., 2025).

Recent work pushes increasingly capable models onto this tier: lightweight edge autoencoders reconstruct and analyse raw Far-Edge telemetry locally, avoiding the Cloud round-trip (He et al., 2026), and quantised deep Q-self-attention transformers run on localised hardware for high-accuracy tasks such as real-time PM2.5 prediction (Balasubramanian et al., 2026). Such models, however, are optimised for data-stream prediction rather than operational resilience: if the hosting node loses power or is partitioned, the service simply terminates. Giving edge nodes local intelligence is necessary, but coordinating isolated components during failure remains unresolved.

2.1 Current orchestration methodologies

Two approaches dominate, both using graph-theoretic state representations where nodes are vertices and com-

munication paths are edges. This allows graph neural networks and attention mechanisms to restrict peer-to-peer communication to the most relevant neighbours, minimising overhead (Chen et al., 2025). First, major European initiatives, including aerOS, NEMO, and FLUIDOS, develop continuum-spanning Meta-Operating Systems that expose heterogeneous infrastructure as a single virtualised resource pool (Gkonis et al., 2026). NEMO’s reference architecture identifies frequent node arrival and departure as a core challenge demanding adaptive management (Chochliouros et al., 2023), though this churn is addressed primarily as a membership and resource-discovery concern, which differs in scope from the simultaneous cross-tier collapse considered here.

Second, the field has widely pivoted to Deep and Multi-Agent Reinforcement Learning (DRL/MARL) for dynamic scheduling (Liu et al., 2025; Zhang et al., 2026). Recent work removes its practical bottlenecks: bi-level and hierarchical designs contain the action space during task offloading (Babaiyan et al., 2026), neuro-symbolic frameworks pair neural optimisation with symbolic constraints to fit edge resource limits (Abbass et al., 2026), and proactive migration absorbs sudden load spikes (Alqerm et al., 2026). These methods scale from single-agent scheduling to multi-operator settings through game-theoretic negotiation (Wang et al., 2026) and cooperative stochastic games (Kuchuk et al., 2026). Yet these approaches are predominantly evaluated under conditions of routine load and mobility, with limited testing against simultaneous cross-tier infrastructure collapse. While hierarchical DRL excels at routing through a healthy continuum, its ability to recover services when the infrastructure itself collapses has not been demonstrated under compound failure conditions.

2.2 The transition to semantic AI and its limits

Mathematical optimisation presumes a well-specified objective and state space, so it is poorly suited to failures where the conflict is which objective applies rather than how to satisfy a fixed one. This limitation has motivated recent research on LLMs and agentic architectures. In orchestration, AgentEdge employs a simulate-before-acting validation step to contain unsafe actions (Gort et al., 2025), while SORA-ATMAS uses policy-aligned governance and high-risk fallbacks to coordinate complex urban services safely (Antuley et al., 2026). In diagnosis, frameworks such as BANS (Xiong et al., 2025) and IntentContinuum (Akbari et al., 2025) employ LLMs to perform root-cause analysis over structured telemetry.

These diagnostic systems are notable because they demonstrate that system-level decoupling is achievable by pairing lightweight edge models for low-latency detection with robust Cloud-tier models for heavy se-

semantic reasoning. A common limitation of these integrations is their evaluation environment. They are predominantly tested against software-level faults, such as bottlenecks or service-level objective (SLO) violations, all on stable physical infrastructure. How these state-of-the-art semantic frameworks perform under the stress of a degrading physical topology remains insufficiently characterised.

2.3 Open challenges

Beyond these evaluation-environment limits, three open challenges limit the evidence for operational resilience. The first is a predominant focus on *steady-state conditions* within current research. For instance, a review of self-adaptive Cloud studies found that availability and stability were the primary goals in only about 9% of the reviewed works (Angelis & Kousiouris, 2025). Consequently, established approaches rely on single-failure models that cannot dynamically restructure system topology during concurrent disruptions.

Second, the *limited generalisation* of reinforcement learning: RL offers limited semantic reasoning under novel conditions. DRL policies tend to overfit their training environments and generalise poorly (Kirk et al., 2023), and simultaneous degradation across multiple tiers is precisely the kind of out-of-distribution state that exposes this weakness. The core difficulty is novelty rather than priority: priority can be encoded as a numerical reward weight, whereas structural novelty is harder to represent in a fixed scalar objective. A policy may observe a symptom, such as a latency spike on a saturated queue, without representing why the spike arose or which cross-domain dependencies interact. Furthermore, the multi-agent credit-assignment problem makes it difficult to attribute a systemic outcome to the local decision that produced it (Foerster et al., 2018), complicating post-event attribution and audit (T. Xu et al., 2026).

Third, the *empirical gap and the Opaque Autonomy Paradox* identified by Caldeira et al. (2026), in which increasing an autonomous system's capability tends to erode the transparency and auditability that high-stakes deployment demands. While recent frameworks propose LLM-based self-healing edge architectures, the operational latency overhead and containment effect of semantic reasoning under physical stress remain insufficiently characterised across this line of work (R. Xu et al., 2024). Consequently, running full LLMs on constrained edge nodes is often impractical, and municipal operators cannot accept opaque restructuring of critical services, which together motivate separating where reasoning happens from where action is taken. Pre-execution validation approaches that simulate recovery plans before committing them (Zhao & Han, 2025) point towards auditable deployment but have been tested only on Cloud fault-injection, not true cross-tier collapse. Published evidence on how an intelligent or-

chestrator behaves under compound failure therefore remains limited, motivating a failure-first paradigm and a decoupled architecture.

3 Agentic middleware architecture

To achieve operational resilience under cascading physical failures, this paper proposes the integration of an active, semantic orchestration layer over the conventional urban deployment model. This three-tier agentic middleware is designed to overcome the structural limitations of reactive frameworks by dynamically restructuring component interactions during infrastructure collapse.

3.1 The three-tier Cloud-Edge continuum

The smart-city environment is modelled as a three-tier physical continuum of heterogeneous devices, from Far-Edge microcontrollers to Cloud servers. The baseline AS-IS architecture (Fig. 1) relies on static or reactive orchestrators. It reproduces the failure chain of the storm scenario: passive Far-Edge gateways, no peer-to-peer coordination at the Smart-Edge, and an orchestrator rendered unreachable during a Cloud partition.

The proposed TO-BE middleware (Fig. 2) addresses these limitations by reconfiguring component interactions as the failure unfolds. Gateways detect missing publications through localised heartbeat timeouts and, where applicable, the Last Will and Testament (LWT) feature of Message Queuing Telemetry Transport (MQTT); lightweight, sensor-specific integrity checks flag telemetry that remains available but violates plausibility or drift constraints. Detection is represented in simulation by a class-specific delay between fault occurrence and visibility to the local agent. Intercepted anomalies then reach the Smart-Edge's deterministic heuristic policies, which resolve isolated faults through peer-to-peer (P2P) coordination (e.g., migrating stateless tasks to a healthy neighbour when CPU utilisation exceeds a threshold).

Each Smart-Edge cluster agent governs a defined geographic zone, providing inter-district failure isolation and keeping routine decisions local to minimise LLM invocation. When a cascade exceeds local capacity, a deterministic Escalation Score triggers the Cloud-tier reasoning orchestrator, which is architecturally modular and, as a proof of concept, instantiated with a quantised, open-weight LLM on a municipality-operated on-premises server.

3.2 System state vector and local heuristics

To evaluate system health locally without continuously streaming raw telemetry to the Cloud, each Smart-Edge agent periodically updates a multidimensional state vec-

tor S by aggregating local node telemetry and anomaly flags intercepted from the Far-Edge gateways:

$$S = \{F_c, D_{cs}, N_i, C_h, T_s\}. \quad (1)$$

Its components are: the concurrent fault count F_c (active, non-isolated faults in the zone); the cross-service dependency depth D_{cs} (maximum topological depth of cascading impact across intersecting systems); a novelty index $N_i \in [0, 1]$ measuring structural dissimilarity of the current failure pattern against a local cache of resolved anomalies; a local offload-capacity score $C_h \in [0, 1]$ derived from remaining resource margins on adjacent healthy nodes; and a tier span $T_s \in \{1, 2, 3\}$, the highest tier the failure has reached (1 = Far-Edge, 2 = Smart-Edge, 3 = Cloud). Representing local health as an explicit, low-dimensional feature vector rather than a latent embedding follows the rationale of explainable feature learning (Vlahek & Mongus, 2023) and makes the escalation trigger easier to inspect. It does not, by itself, explain the LLM’s internal action-selection process.

Because the state vector mixes continuous and categorical features, N_i is computed using Gower’s distance, which handles mixed types natively. Each district cache contains one entry for every eligible calibration signature successfully contained by the heuristic-only baseline. It has no separately imposed capacity limit, as its realised size and lookup cost are reported with the experimental configuration. N_i is the mean Gower distance between the current signature (F_c, D_{cs}, C_h, T_s) and its k nearest eligible cache entries, where k is a fixed calibration constant. The cache is populated only from calibration cases the heuristic layer successfully contained, and no case contributes to scoring itself, preventing leakage. After model selection, the caches are reconstructed from all calibration seeds and frozen for evaluation, with no online update or eviction. If a district has fewer than k eligible entries, the remaining neighbours are drawn without duplication from a pooled calibration cache built under the same eligibility rules.

3.3 Deterministic escalation and constrained reasoning

To address the Opaque Autonomy Paradox, a deterministic trigger governs the transition to Cloud-tier reasoning, and outputs are strictly constrained.

The Escalation Score is a sigmoid over a linear combination of the state-vector features, with weights β and intercept β_0 calibrated from labelled simulation data before evaluation:

$$ES(S) = \sigma(\beta_0 + \beta^\top S), \quad ES \in (0, 1), \quad (2)$$

where $\sigma(x) = 1/(1+e^{-x})$ is the logistic function and S is the state vector of Eq. (1). Calibration labels are generated in simulation under the heuristic-only baseline. A state sampled at time t is labelled 1 if, during a fixed

look-ahead horizon, at least 1% of the safety-critical tasks issued in that horizon miss their class-level application deadlines; otherwise, it is labelled 0. The horizon is selected to be long enough to capture service breaches that timely escalation could still prevent, and both the horizon and the 1% threshold are fixed before labelling begins. The operating threshold is then selected through seed-grouped cross-validation, with all state vectors originating from the same simulated scenario assigned to the same fold. The threshold is chosen to minimise unnecessary escalations while detecting a pre-specified minimum proportion of states that lead to service impact and remaining within the measured capacity of the shared Cloud-tier orchestrator. Calibration and evaluation use disjoint scenario seeds, and the fitted parameters and operating threshold are frozen before the final evaluation.

Escalation is governed by a hysteretic trigger on the action $a_t \in \{0, 1\}$ at update step t :

$$a_t = \begin{cases} 1 & \text{if } ES_t \geq \tau_{\text{high}}, \\ 0 & \text{if } ES_t \leq \tau_{\text{low}}, \\ a_{t-1} & \text{otherwise.} \end{cases} \quad (3)$$

with $0 < \tau_{\text{low}} < \tau_{\text{high}} < 1$.

The Cloud orchestrator is invoked only once ES rises above τ_{high} and is released only once it falls below τ_{low} ; the deadband $[\tau_{\text{low}}, \tau_{\text{high}}]$ suppresses chattering when the state hovers near the boundary.

Once triggered, the agent transmits state vector, the relevant dependency subgraph, the active failure set, and a deterministically generated set of currently feasible actions. Two decoding constraints are applied to reduce sampling randomness and prevent structurally invalid outputs:

- **Greedy decoding:** The municipality-hosted LLM uses greedy decoding with the temperature set to zero.
- **Structural enforcement via constrained decoding:** The orchestrator enforces a strict JSON schema through token-level grammar constraints using the *Outlines* library. Outputs are restricted to valid escalation identifiers, target-agent identifiers, and the four defined orchestration primitives: MIGRATE, SHED, RESTART, and NO_OP.

The schema is regenerated from the current feasible-action set, so the grammar admits only action–target combinations whose preconditions hold at the start of the decision cycle. This provides structural and target validity at generation time, but it does not guarantee that the selected action is optimal or beneficial. Greedy decoding with temperature zero reduces sampling variability but is not treated as a guarantee of bitwise reproducibility across different model, hardware, or software configurations.

Because the system state may change during inference, the selected action is revalidated immediately before commitment. An action whose preconditions no

longer hold is rejected and logged with the corresponding reason code. Each decision cycle records the state, Escalation Score, feasible-action set, generated action, validation result, and committed action. These records make the escalation pathway reconstructible after an event without claiming to provide a faithful explanation of the LLM’s internal reasoning. Action quality is assessed empirically through the fault-containment score (FCS) and the secondary service metrics. To bound the principal technical risk that Cloud-tier inference may exceed operational limits, the middleware integrates two mitigation strategies. First, Smart-Edge agents use an asynchronous local fallback: while awaiting the LLM response, they continue bounded local coordination and may shed best-effort services to prevent resource exhaustion. Second, every Cloud-tier decision cycle has a control-plane time-to-live (TTL), denoted t_{deadline} . If no valid action is committed before this deadline, the cycle ends in a deadline abort and control remains with the local heuristic fallback. The evaluation therefore tests both containment benefit and the frequency and latency cost of Cloud-tier decision cycles.

4 Proposed evaluation framework

Evaluating an architecture built for infrastructure collapse requires moving beyond steady-state testing towards a failure-first paradigm. Grounded in Design Science Research (DSR), the evaluation tests containment performance together with two governance properties relevant to municipal deployment: adherence to service-level agreements (SLAs) and a reconstructible decision pathway. Because intentionally triggering cascading failures on operational urban infrastructure is unsafe and unethical (Q. Li & Gao, 2026), the architecture is designed to be evaluated through discrete-event simulation mapped to open-data-derived urban dependency graphs.

4.1 Research hypotheses

To quantify the trade-off between the cost of failure and the cost of intelligence, the framework defines the following planned evaluation hypotheses; they are predictions to be tested rather than demonstrated findings:

- **H_{1a} (Cloud decision component):** Under compound failures, the full middleware achieves a higher FCS than a deterministic topology-aware Cloud-tier planner operating through the same escalation pathway.
- **H_{1b} (Complete pathway):** Under compound failures, the full middleware achieves a higher FCS than the heuristic-only strategy.
- **H_2 (Complexity scaling):** The Full-B4 FCS difference is more strongly associated with cross-service dependency depth than with the raw number of faults.

- **H_3 (Cost of intelligence):** In compound-failure scenarios where the Cloud-tier pathway remains reachable, fewer than 1% of initiated decision cycles end in a deadline abort.

All strategies are evaluated on matched, seed-replayed scenarios. H_{1a} isolates the Cloud-tier decision component through the Full-B4 contrast, H_{1b} evaluates the complete escalation pathway against B3, H_2 models the structural conditions under which the Full-B4 difference changes, and H_3 evaluates adherence to the control-plane deadline.

4.2 Empirical simulation grounding

To ensure the simulated environment reflects operational reality, the framework maps the fault taxonomy (silent sensor dropout, drift, link partition, and node overload) directly to empirical failure rates derived from real deployments. These distributions incorporate sensor data-integrity faults (Talayoglu et al., 2025), edge low-power wide-area network (LPWAN) link losses (Yasmin et al., 2020), backbone and Cloud-network failures (Talluri et al., 2026), and Cloud-tier server hardware failures (Xie et al., 2025). The service dependency graph is derived from a published open-data urban infrastructure model (Di Pietro et al., 2025). By applying Monte Carlo sampling over these empirical distributions and the urban topology, the simulation generates realistic, combinatorial fault intersections.

Testing an LLM orchestrator against latency deadlines requires incorporating measured inference delays into simulated control-plane timing. The simulator of choice is Yet (not) Another Intelligent Fog Simulator (YAIFS), developed by Lera and Guerrero (2026). Its Model Context Protocol (MCP) integration lets external LLM agents observe system state and inject orchestration actions. The evaluation records serialisation, network transfer, queueing, inference, validation, and dispatch time for each decision cycle.

4.3 Quantifying the core trade-off

The primary cost-of-failure metric is the FCS, adapted from Mohapatra (2025):

$$\text{FCS} = \frac{N_{\text{isolated}}}{N_{\text{total}}} \times (1 - \delta), \quad \text{FCS} \in [0, 1], \quad (4)$$

where N_{total} is the injected ground-truth fault count, N_{isolated} is the number of faults contained within their originating district without degrading a neighbouring district or higher tier, and δ is the propagation-severity coefficient. Mohapatra, 2025 specifies the $(1 - \delta)$ form but defines δ only qualitatively; we operationalise it on $[0, 1]$ as follows. For multiple injected origins, the denominator of δ is the union of nodes reachable from those origins in the pre-failure dependency graph; injected origin nodes are excluded and overlapping reachable

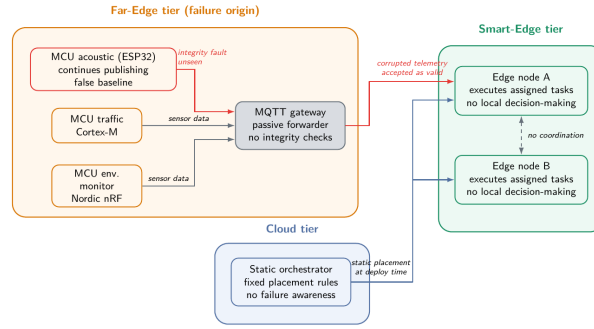


Figure 1. Baseline (AS-IS) architecture: the lack of active edge-tier coordination allows compound physical failures to cascade across tiers.

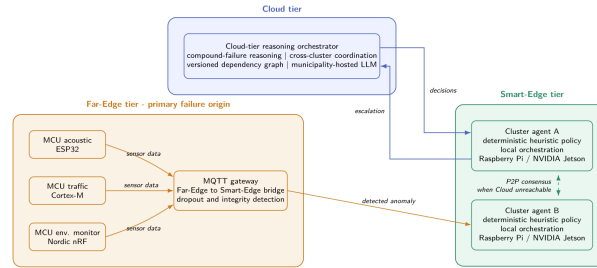


Figure 2. Proposed (TO-BE) three-tier agentic middleware, showing the escalation path from Far-Edge gateways to the Cloud-tier LLM orchestrator.

sets are counted once. The numerator contains the subset that remains degraded at the stabilised end of the scenario. The $(1 - \delta)$ multiplier penalises strategies that allow uncontained faults to trigger widespread structural collapse.

To reflect mixed criticality, intentionally shed best-effort services are excluded from the affected-node set used for δ , but shedding is reported separately through shed-instance count and cumulative shed-service duration. Successfully migrated or rerouted services that remain operational are not counted as degraded; only unresolved secondary propagation contributes. A node counts as degraded only if an offline, unreachable, or task-breach condition persists for a full telemetry interval.

Secondary cost-of-failure metrics are critical-service uptime and task completion; cost-of-intelligence metrics are decision-cycle p_{95}/p_{99} latency, Cloud invocation, validation-rejection, and deadline-abort rates. End-to-end critical-service latency is reported separately.

4.4 Adversarial evaluation via threshold evasion

The protocol also includes a slow-onset threshold-evasion stress test. Faults are temporally staggered and constrained to states that keep the Escalation Score below τ_{high} until a service breach is imminent. This “Boiling Frog” cascade tests whether gradual structural degradation can evade the calibrated gate and identifies conditions under which escalation should occur earlier.

4.5 Baseline comparisons

The evaluation compares five strategies arranged as a capability ladder, each adding one mechanism over the previous. The ladder fixes the expected capability ordering for the hierarchy check, while B3 and B4 additionally function as targeted ablations of the full system:

- **B1 (Static orchestration):** Fixed service placement with no dynamic adaptation, included as the non-adaptive lower bound of the ladder.
- **B2 (Reactive rule-based):** Threshold-triggered single-node recovery, modelling traditional self-healing (Boluda-Prieto et al., 2026), included to locate where single-node recovery breaks down under compound collapse.
- **B3 (Heuristic-only ablation):** The proposed local deterministic layer with the Cloud pathway disabled, using resource-aware, TTL-bounded gossip and nearest-neighbour offloading (Y. Li et al., 2026).
- **B4 (Deterministic Cloud-tier planner):** The complete escalation pathway with the LLM replaced by a deterministic topology-aware greedy planner operating over the same payload, feasible-action set, Cloud resource, and decision deadline.

B3 tests the value of the complete escalation pathway, whereas B4 isolates the value added by the LLM decision component from the value of Cloud reachability, city-wide structural context, and escalation itself.

5 Discussion and conclusion

The proposed design separates fast edge heuristics from selective Cloud reasoning during compound smart-city failures. Its Escalation Score, constrained schema, pre-execution validation, and decision logging support a reconstructible escalation pathway, while the failure-first framework compares the *cost of failure* with the *cost of intelligence*.

The architecture's central trade-off is not simply whether Cloud-tier reasoning is more capable than local rules, but whether the structural novelty of a failure justifies delayed action from the Cloud tier. The planned hypotheses make that trade-off falsifiable. A Full-B4 FCS improvement accompanied by frequent deadline aborts would indicate that LLM-based reasoning can help containment but is unsuitable for the most time-critical decisions; acceptable decision-cycle latency without improved containment would suggest that the LLM adds little beyond deterministic Cloud planning. H_2 is especially important for scope: an advantage associated with dependency depth rather than fault count would support the smart-city motivation based on cross-operator dependencies and correlated physical events. Negative results would still identify conditions under which Cloud escalation or LLM-based planning should be disabled.

This design contribution reports no empirical results; its hypotheses remain planned tests. Next, the Escalation Score will be calibrated on held-out data and evaluated in YAIIS against the four baselines, including threshold-evasion cascades and measured Cloud-tier decision delays. Later work may examine lightweight edge RL and generate-and-score verification of orchestration actions (cf. Gort et al., 2025; Zhao & Han, 2025).

Acknowledgements

This work is fully supported under FOI-IIP-2025-13.

References

- Abbass, W., Abbas, N., & Majeed, U. (2026). Neuroscale: Scalable and constraint-aware neuro-symbolic optimization for edge-centric IoT systems. *The Journal of Supercomputing*, 82(3), 127.
- Akbari, N., Grundy, J., Cheema, A., & Toosi, A. N. (2025). IntentContinuum: Using LLMs to support intent-based computing across the compute continuum. *2025 IEEE International Conference on Web Services (ICWS)*, 573–583.
- Alqerm, I., Cheng, N., & Pan, J. (2026). REPA: Resilient and proactive resource allocation for last-mile heterogeneous edge-IoT systems. *IEEE Internet of Things Journal*, 13(2), 1988–2005.
- Angelis, A., & Kousiouris, G. (2025). An overview on the landscape of self-adaptive cloud design and operation patterns. *Future Internet*, 17(10), 434.
- Antuley, U., Siddiqui, S., Hameed, S., Arif, W., & Shah, S. A. (2026). SORA-ATMAS: Adaptive trust management and multi-LLM aligned governance for future smart cities. *Knowledge-Based Systems*, 337, 115403.
- Babaiyan, V., Bushehrian, O., & Javidan, R. (2026). A bi-level mobility-aware deep reinforcement learning approach for fault-tolerant task offloading in vehicular edge-cloud computing. *Scientific Reports*, 16(1), 11703.
- Balasubramanian, C., Kareemullah, H., Arunkumar, R. S., Kumar, P. S., & Revathi, S. M. (2026). Multi-objective deep Q-self-attention transformer for PM2.5 prediction on edge devices. *GeoInformatica*, 30(1), 11.
- Belcastro, L., Marozzo, F., Orsino, A., Talia, D., & Trunfio, P. (2026). Navigating the edge-cloud continuum: A state-of-practice survey. *IEEE Access*, 14, 40622–40647.
- Boluda-Prieto, M., Mateo-Casali, M. A., Fraile, F., & Alarcon, F. (2026). Resilient edge-to-cloud architecture with self-healing and self-correcting mechanisms for industrial data continuity. *Computers & Industrial Engineering*, 213, 111795.
- Caldeira, I. D. B., Choren, R., & Salles, R. (2026). The opaque autonomy paradox in sovereign and intelligent SD-WAN: A systematic literature review. *Computer Networks*, 282, 112246.
- Chaturvedi, P., Ahmad, S., & Mewada, A. (2026). A comprehensive survey on fog computing: Architectures, techniques, challenges, and future directions. *Archives of Computational Methods in Engineering*.
- Chen, S., Li, D., & Mohajer, A. (2025). QoS-aware multi-agent DDPG for adaptive edge service distribution in intelligent wireless communication networks. *Ain Shams Engineering Journal*, 16(9), 103543.
- Chochliouros, I. P., Pages-Montanera, E., Alcázar-Fernández, A., Zahariadis, T., Velivassaki, T.-H., Skianis, C., Rossini, R., Belesiotti, M., Drosos, N., & Bakiris, E. (2023). NEMO: Building the next generation meta operating system. *Proceedings of the 3rd Eclipse Security, AI, Architecture and Modelling Conference on Cloud to Edge Continuum*, 1–9.
- Di Pietro, A., Cavedon, F., Stergiopoulos, G., & Rosato, V. (2025). Modeling fault dynamics of interdependent assets in complex urban environments using open-data. *SN Computer Science*, 6(6), 564.
- Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., & Whiteson, S. (2018). Counterfactual multi-agent policy gradients. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).
- Gkonis, P. K., Giannopoulos, A., Nomikos, N., Sarakis, L., Nikolakakis, V., Patsourakis, G., & Trakadas, P. (2026). A survey on the computing continuum and meta-operating systems: Perspectives, architectures, outcomes, and open challenges. *Sensors*, 26(3), 799.

- Gort, B., Kibalya, G. M., & Antonopoulos, A. (2025). *AgentEdge: Agentic AI for service orchestration in the edge-cloud continuum* [TechRxiv preprint].
- He, J., Zheng, Y., Lei, S., Liao, F., & Wu, L. (2026). Cyber-physical intrusion detection for coordinated renewable energy and storage systems. *Array*, 30, 100765.
- Kirk, R., Zhang, A., Grefenstette, E., & Rocktäschel, T. (2023). A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76, 201–264.
- Kuchuk, H., Kosenko, N., Kuchuk, N., Gopejenko, V., & Kosenko, V. (2026). Adaptive resource allocation method for the mobile fog layer of high-density industrial internet of things in industry 5.0 networks. *Innovative Technologies and Scientific Solutions for Industries*, (1(35)), 65–78.
- Lera, I., & Guerrero, C. (2026). *YAFIS: Yet (not) another intelligent fog simulator* [arXiv:2604.19181].
- Li, Q., & Gao, K. (2026). Beyond the Comfort Zone: A Review and Gap Analysis of Fuzzing in Smart City IoT Ecosystems. *Information*, 17(3), 218. <https://doi.org/10.3390/info17030218>
- Li, Y., Liu, F., Chang, T.-Y., Hsu, C.-H., & Venkatasubramanian, N. (2026). Microservice provisioning and event-driven adaptation in heterogeneous IoT settings. *Pervasive and Mobile Computing*, 118, 102195.
- Liu, Q., Yang, J., & Yan, Z. (2025). Dynamic resource orchestration in edge computing environments using multi-agent reinforcement learning. *Knowledge and Information Systems*, 67(10), 9363–9383.
- Mohapatra, H. (2025). A comprehensive review on urban resilience via fault-tolerant IoT and sensor networks. *Computers, Materials & Continua*, 85(1), 221–247.
- Talayoglu, B., Vande Velde, J., & Da Silva, B. (2025). Lightweight AI for Sensor Fault Monitoring. *Electronics*, 14(22), 4532. <https://doi.org/10.3390/electronics14224532>
- Talluri, S., Niewenhuis, D., Chu, X., Kyselica, J., Cetin, M., Balgavy, A., & Iosup, A. (2026). Cloud Uptime Archive: Open-Access Availability Data of Web, Cloud, and Gaming Services. *IEEE Transactions on Parallel and Distributed Systems*, 37(5), 1136–1152. <https://doi.org/10.1109/TPDS.2026.3673519>
- Vlahek, D., & Mongus, D. (2023). An efficient iterative approach to explainable feature learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(5), 2606–2618. <https://doi.org/10.1109/TNNLS.2021.3107049>
- Wang, Y., Guo, S., Pan, Y., Su, Z., Chen, F., Luan, T. H., Li, P., Kang, J., & Niyato, D. (2026). Internet of agents: Fundamentals, applications, and challenges. *IEEE Transactions on Cognitive Communications and Networking*, 12, 4476–4501.
- Wei, Q., Ning, H., Shi, F., Zhu, T., Ding, J., Derhab, A., & Daneshmand, M. (2026). AGI-enabled solutions for service explosion in IoX: A cyber-physical-social-thinking perspective. *ACM Computing Surveys*, 58(8), 1–32.
- Xie, H., Zhou, M., Yu, Q., Yu, J., Sheng, Z., Xie, H., & Lian, D. (2025). M²S-MFP: A Multi-Scale and Multi-Level Memory Failure Prediction Framework for Reliable Cloud Infrastructure. *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, 5083–5094. <https://doi.org/10.1145/3711896.3737243>
- Xiong, Z., Meng, S., Yao, S., & Luo, L. (2025). Fault tolerant and intelligent reliability assurance in cloud-edge systems. *2025 12th International Conference on Dependable Systems and Their Applications (DSA)*, 156–162.
- Xu, R., Nagothu, D., & Chen, Y. (2024). AR-Edge: Autonomous and Resilient Edge Computing Architecture for Smart Cities. In Y. Chen & R. Xu (Eds.), *Edge Computing Architecture - Architecture and Applications for Smart Cities*. IntechOpen. <https://doi.org/10.5772/intechopen.1005876>
- Xu, T., Zhang, D., Mitra, K., & Hruschka, E. (2026). Verification-Aware Planning for Multi-Agent Systems. *EACL 2026*. <https://doi.org/10.18653/v1/2026.eacl-long.353>
- Yasmin, R., Mikhaylov, K., & Pouttu, A. (2020). LoRaWAN for Smart Campus: Deployment and Long-Term Operation Analysis. *Sensors*, 20(23), 6721. <https://doi.org/10.3390/s20236721>
- Zhang, G., Zhang, Q., Feng, L., Zhou, F., Donta, P. K., & Dustdar, S. (2026). Task-aware collaborative inference and fine-grained DNN partitioning in MEC networks. *IEEE Transactions on Mobile Computing*, 1–17.
- Zhao, Y., & Han, Z. (2025). Safe and adaptive cloud healing: Verifying LLM-generated recovery plans with a neural-symbolic world model. *Proceedings of the 7th International Academic Exchange Conference on Science and Technology Innovation (IAECST)*, 89–94.