future internet

MDPI

Article

Distributed Artificial Intelligence for Agentic Intent-Based Networking in Mobile Networks

Iacovos Ioannou [1,*], Susanne Naegele-Jackson [2], Sonja Filiposka [3], Vincent Burkard [2], Michael Georgiades [1], Michael Mackay [4] and Vasos Vassiliou [5,6]

[1] Department of Computer Science, Intelligent Systems Laboratory, Neapolis University Pafos, 8042 Pafos, Cyprus; m.georgiades@nup.ac.cy
[2] Erlangen Regional Computing Center (RRZE), Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 91058 Erlangen, Germany; susanne.naegele-jackson@fau.de (S.N.-J.); vincent.burkard@fau.de (V.B.)
[3] Faculty of Computer Science and Engineering (FINKI), Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia; sonja.filiposka@finki.ukim.mk
[4] School of Computer Science and Mathematics, Liverpool John Moores University, Liverpool L3 3AF, UK; m.i.mackay@ljmu.ac.uk
[5] Department of Computer Science, University of Cyprus, 1678 Nicosia, Cyprus; vasosv@ucy.ac.cy
[6] CYENS Centre of Excellence, 1016 Nicosia, Cyprus
* Correspondence: iakovos.ioannou@nup.ac.cy

Abstract

Through intent-based networking (IBN), network objectives are specified declaratively, yet in mobile networks their realisation spans administrative domains that compete for coupled radio, transport and compute resources but cannot share local models and telemetry. A distributed artificial intelligence (DAI) controller with a Belief-Desire-Intention-eXecution (BDIX) cycle is presented, centred on shared-space coordination. The local decision of each domain agent is mapped through G_i into a common p-dimensional resource space, so that only resource contributions and a common price vector cross domains, after a minimum-slack feasibility certificate and behind a trust-admission gate. Typed refinement, deficit feedback and conflict mediation support it. In a coordination-layer benchmark (4 to 64 domains, twenty seeds), the full-information objective is matched to the reported numerical precision without disclosure of local models (cost gap 0.00%), whereas, at the principal $\beta = 0.60$ setting, aggregate-only allocation loses 2.27% to 3.83% and equal split loses 25.21% to 37.67%. When 12.50% to 25.00% of the domains understate their contributions fivefold, honest-domain cost rises by 24.48% to 50.95% without the gate and by at most 2.26% with it, while mild falsification is not reliably detected by the trust gate, leaving containment dependent on the delegated resource bounds. On the single-domain analytical plant (twenty hold-out seeds), the controller attains the joint-highest displayed score (0.80), the lowest mean latency and significantly higher scores than seven of eleven comparators; end-to-end workflow execution is demonstrated separately by bridge dry runs. Shared-space coordination is thus the allocation of choice where local models cannot be shared.

Keywords: intent-based networking; distributed artificial intelligence; multi-agent systems; mobile networks; O-RAN; network slicing; distributed optimisation; Lyapunov control; shared-space coordination; zero-touch management

Check for updates

Academic Editors: Diego Vergara and Pablo Fernández-Arias

Received: 21 August 2026
Revised: 24 September 2026
Accepted: 28 September 2026
Published: 1 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction and Motivation

Mobile networks are becoming simultaneously more autonomous and more fragmented as they evolve towards sixth-generation (6G) and beyond. They now combine

fifth-generation (5G) core service-based functions, disaggregated or cloudified radio access network (RAN) elements, multi-access edge computing (MEC), network slicing, analytics functions such as the Network Data Analytics Function (NWDAF) and increasingly open or programmable control planes such as Open Radio Access Network (O-RAN). These developments improve flexibility and observability, but they also create an operational challenge: multiple independent control loops, optimisation tools and policy consumers must coexist while honouring heterogeneous service intents [1–3].

Intent-based networking (IBN) addresses part of this problem by giving operators and customers a declarative way to specify what should be achieved rather than how to configure the network [4,5]. However, the presence of a northbound intent interface alone does not solve realisation. Once a high-level intent enters the management system, it must still be interpreted, decomposed, negotiated against resource availability, translated to domain-specific policies, enforced through closed loops and continuously assured as mobility, load variation and failures perturb the underlying network.

The difficulty is therefore not a shortage of automation but an excess of uncoordinated automation: individually reasonable control loops may pursue incompatible objectives on inconsistent views of the network, and an intent must survive this environment, which makes realisation a systems problem rather than an interface problem.

A purely centralised controller is attractive for governance and visibility, but mobile-network control is naturally distributed for at least five reasons. Geographical distribution matters because the RAN, transport, MEC and core domains are physically dispersed and often latency-sensitive. Administrative heterogeneity matters because different domains may be owned by different vendors, tenants or even different operators. Timescale heterogeneity matters because spectrum and scheduling decisions occur far faster than slice-envelope or business-intent updates. Data locality and privacy matter because raw telemetry can expose mobility patterns, service composition and commercially sensitive topology. Robustness matters because centralised orchestration forms a single concentration point for model errors, stale state or compromise. Distributed Artificial Intelligence (DAI) is therefore adopted to support local decision-making under shared constraints. This choice is motivated by earlier agent-based designs for mobile-network control [6]. The right abstraction is not "one model for the whole network" but a federation of bounded agents that can reason locally, coordinate globally and remain constrained by typed policies and safety invariants.

Because control is distributed, the coordination between agents must be made explicit and provable rather than assumed; for this reason a distributed optimisation formulation, a convergence condition and a trust-aware coordination algorithm are developed in the sections that follow.

Recent network-management work with large language models (LLMs) and agentic orchestration shows promise for workflow decomposition, tool use and operator interaction [7,8]. Yet mobile networks impose requirements that generic agentic systems rarely face: deterministic safety envelopes, explicit service-level agreement (SLA) objectives, rollback paths, auditability, bounded control rights and coexistence with formal telecom standards. A narrow definition of agentic is therefore adopted in this paper. An agent is not an unrestricted autonomous actor. It is a policy-governed, capability-bounded decision-maker that consumes typed intents, acts only through explicit interfaces and tools, remains inside a trust and safety envelope, shares only the information that its peers require and can be overridden by a local safety layer.

This narrow definition is what makes the later guarantees meaningful: a policy-governed agent can be checked against an explicit objective, a capability-bounded agent has a delimited action scope that admits formal analysis, typed inputs and outputs keep

the optimisation and actuation paths reproducible, minimal sharing keeps the trust surface small and the local safety override prevents a reasoning failure from becoming an unsafe actuation.

The open choices are explicit by design. A guiding requirement for this work is that unspecified deployment details are left as explicit options. Accordingly, one mobility model, one network size or one key performance indicator (KPI) target set is not hard-coded. Instead, analytical or trace-driven mobility, small private, metro-scale or large multi-domain deployments, enhanced mobile broadband (eMBB), ultra-reliable low-latency communication (URLLC), massive Internet of Things (IoT), energy-aware or security-intensive intent portfolios and static, adaptive or learned local controllers are all permitted as architectural options. That design choice keeps the contribution reusable as a blueprint rather than a single-case study.

The contribution of this work is focused on one mechanism, namely shared-space coordination among domain agents whose local models and telemetry cannot be exchanged. The complete controller is denoted DAI-BDIX because a Belief-Desire-Intention-eXecution (BDIX) cycle is formed by extending the classical agent cycle with an explicit eXecution stage that binds selected intentions to governed network actions. The remaining ingredients, namely typed intent objects, hierarchical intent refinement, Lyapunov-guided online execution and finite potential games, are separately grounded in established theory and are retained as the supporting stack within which the coordination mechanism operates. A common formulation is kept in which their assumptions and interfaces are made explicit, so that an intent predicate, an optimisation variable, a virtual queue, a control action and a trust score all refer to the same underlying mobile-network state.

The specific problem is that existing intent-realisation schemes either assume a coordinator that holds every local model, which administrative separation and data locality rule out, or allocate coupled resources from aggregate declarations alone, which ignores the marginal value of a resource to each domain. Three constructions distinguish the proposed mechanism from prior art. First, each domain agent maps its heterogeneous local decision through a shared-space map G_i into a common p-dimensional resource space, so that only resource contributions and a common dual vector cross a domain boundary, while the local objective, the local feasible set and the telemetry from which they are estimated remain private. Second, the Phase-I minimum-slack certificate is separated from the operating objective, so that joint infeasibility is detected before economic coordination and cannot be masked by a finite penalty. Third, trust admission is applied to the exchanged contributions themselves, so that a falsified contribution can be excluded before it distorts the common price. The guarantees that the constituent methods already carry, namely fulfilment preservation under sound refinement, convergence of the primal-dual hybrid-gradient iteration, the $O(1/V)$ drift-plus-penalty bound and exact-potential convergence, are instantiated for this formulation and are not claimed as new results. The construction is mapped onto concrete 3rd Generation Partnership Project (3GPP), European Telecommunications Standards Institute (ETSI) and O-RAN actuation points so that it remains realisable on systems that operators are already deploying.

The claim of the manuscript is accordingly a single one: shared-space coordination matches the allocation of a full-information coordinator to the reported numerical precision without disclosure of local models and it remains protected against falsified contributions when trust admission is active. The evidence for this claim is provided by the coordination-layer benchmark reported in the evaluation, while it is shown by the analytical campaign, evaluated in depth on twenty hold-out seed blocks against eleven comparator configurations and six ablations and by the archived legacy bridge dry runs that the mechanism is embedded in the complete stack without loss: the joint-highest displayed score, the

lowest mean latency on the plant and significant gains over seven of eleven comparators are attained by the complete controller, while additionally providing trust admission and certified feasibility, neither of which is provided by any comparator. Accordingly, the following contributions are made in this manuscript.

1. Shared-space coordination is proposed as the principal contribution: heterogeneous domain agents are coordinated through maps G_i into a common p-dimensional resource space, the Phase-I feasibility certificate is separated from the operating objective and trust admission is applied to the exchanged contributions, so that neither local models nor telemetry leave a domain. Convergence is obtained by instantiating the primal-dual hybrid-gradient result [9] under stated assumptions.
2. The supporting stack is formalised: a standards-aligned DAI architecture across business, service, domain and execution layers, formal definitions of agents, intents, intent hierarchies, network state and mobility models with a unified notation table and a constrained stochastic multi-agent formulation with Phase-I minimum-slack, game-theoretic and control-theoretic surrogates. The known guarantees of fulfilment preservation by composition of sound refinements, the drift-plus-penalty bound [10] and exact-potential convergence [11] are instantiated for this formulation, together with the explicit construction and verification of sound refinement operators.
3. Three explicit algorithms are presented with pseudocode and complexity analysis and security, privacy and trust are integrated directly into the formal control architecture.
4. A coordination-layer benchmark is provided in which the mechanism must make a difference: 4 to 64 administrative domains with non-shareable local models are compared under full-information, shared-space, aggregate-only and equal-split allocation and a contribution-falsification attack is evaluated with trust admission switched on and off.
5. A five-scenario paired system-level simulation campaign is provided for the complete controller, with authored comparators, six internal ablations, full KPI reporting, run-level uncertainty, paired significance analysis and mechanism-specific trust, convergence, churn and safety traces. Recovered legacy bridge records for an ambiguous disaster-recovery intent scenario are retained with their dry-run provenance; the first rank on the bridge endpoint (0.89) is obtained by the current stack executed through the same bridge workflow, whereas the last rank is obtained by the archived legacy pipeline.

The paper map is provided as follows. The motivation and problem setting are established in Section 1. The standards baseline and related work are reviewed in Section 2. The system model, formal definitions, notation and problem formulation are developed in Section 3. The proposed algorithms and their complexity are presented in Section 4. Security, privacy and trust are treated in Section 5. A proof-of-concept evaluation and a reproducibility blueprint are given in Section 6. The discussion is provided in Section 7 and the work is concluded in Section 8.

2. Background, Standards Context and Related Work

In this section, the background, standards context and related work that ground the proposed framework are presented. The 3GPP, ETSI and O-RAN specifications that define intent management, zero-touch closed loops and open radio-access control are first summarised as the baseline against which the proposed functions are mapped. Work on intent modelling, translation and closed-loop realisation is then reviewed, followed by the distributed artificial intelligence and multi-agent control literature on which the agent design rests and by the studies of O-RAN conflict, agentic operations and trust that motivate the safeguards of the framework. The section closes with the research gap that

remains after this review and with the position of the present work relative to the criteria by which the related families are compared.

2.1. Standards Baseline

Recent telecom standardisation provides substantial support for this topic. The Request for Comments (RFC) series includes publications of the Internet Engineering Task Force (IETF) and Internet Research Task Force (IRTF). Core intent-based networking concepts are clarified in the IRTF Informational RFC 9315 [4]. Intent classification is addressed separately in RFC 9316 [12]. In 3GPP, Technical Specification (TS) 28.312 specifies intent-driven management services for mobile networks and formally models intent-related artefacts such as expectations, contexts and handling capabilities [5]. Technical Report (TR) 28.912 extends that work towards enhanced intent-driven management services [13]. Management data analytics is specified in 3GPP TS 28.104 [14]. Artificial intelligence/machine learning (AI/ML) management is addressed in TS 28.105 [15]. Mobile analytics exposure is specified in TS 23.288 [1]. Baseline 5G System (5GS) security remains anchored in TS 33.501 [16].

The second major pillar is provided by ETSI. A reference architecture is defined by Zero-touch network and Service Management (ZSM) [17] and closed-loop enablers are specified in Group Specification (GS) 009-1 [18]. Generic aspects of intent-driven autonomous networks are discussed in Group Report (GR) 011 [19], intent-driven closed loops are specified in GS 016 [20], network digital twins are addressed in GS 018 [21] and agents in autonomous networks are studied in GR 020 [22]. The Experiential Networked Intelligence (ENI) system architecture is defined in GS 005 [23]. Intent-policy gaps are analysed in GR 013 [24], multi-stage translation is specified in GS 033 [25] and conflict detection is specified in GS 034 [26]. AI agents for network slicing are studied in GR 051 [27] and multi-agent frameworks for core networks are studied in GR 056 [28].

The third pillar is provided by O-RAN through concrete open interfaces and application points: public specifications and recent technical-document releases describe the near-real-time (near-RT) RAN intelligent controller (RIC), the E2 application protocol (E2AP) and E2 general aspects and principles (E2GAP), service models, conflict-mitigation support and AI/ML model-management application programming interfaces (APIs) [2,29,30]. Representative documents are summarised in Table 1. IETF and IRTF denote the Internet Engineering Task Force and Internet Research Task Force; TS, TR, GS and GR denote Technical Specification, Technical Report, Group Specification and Group Report, respectively.

Table 1. Representative standards and public specifications relevant to agentic intent-driven mobile networking.

Body	Document	Main Contribution	Relevance Here
IETF/IRTF	RFC 9315, RFC 9316	Shared definitions of intent and intent classes.	Used to type intent objects and refine semantics.
3GPP	TS 28.312, TR 28.912	Intent-driven management services for mobile networks and enhancements.	Main standards anchor for the intent lifecycle and information model.
3GPP	TS 28.104, TS 28.105, TS 23.288	Management analytics and AI/ML lifecycle support.	Underpins analytics, model management and state observability.
3GPP	TS 33.501	5GS security architecture.	Baseline telecom security and trust assumptions.
ETSI ZSM	GS 002, GS 009-1, GR 011, GS 016, GS 018, GR 020	Reference architecture, closed loops, intent-driven loops, network digital twins (NDTs) and agents.	Primary source for closed-loop and agentic governance.

Table 1. *Cont.*

Body	Document	Main Contribution	Relevance Here
ETSI ENI	GS 005, GR 013, GS 033, GS 034, GR 051, GR 056	ENI system architecture, translation, conflict detection and AI-agent studies.	Primary source for translation, conflict and agent design patterns.
O-RAN	Public specs and recent technical releases	RAN intelligent controller (RIC) architecture, E2 service models, KPI and control exposure, conflict-mitigation functionality.	Primary source for realisable RAN-domain actuation points.

Abbreviations: Internet Engineering Task Force (IETF); Internet Research Task Force (IRTF); Request for Comments (RFC); 3rd Generation Partnership Project (3GPP); Technical Specification (TS); Technical Report (TR); European Telecommunications Standards Institute (ETSI); Zero-touch network and Service Management (ZSM); Experiential Networked Intelligence (ENI); Group Specification (GS); Group Report (GR); artificial intelligence (AI); machine learning (ML); fifth-generation (5G) System (5GS); Open Radio Access Network (O-RAN); radio access network (RAN); key performance indicator (KPI).

The standards' references above identify conceptual and functional anchors; they do not imply that the present proof of concept has passed protocol-level conformance testing. The mapping in Table 2 makes this distinction explicit.

Table 2. Mapping of proposed architectural functions to standards artefacts and proof-of-concept status.

Proposed Function	Relevant Artefact	Conceptual Alignment	Status in This Proof of Concept
Intent ingress and lifecycle	3GPP TS 28.312, TR 28.912; RFC 9315/9316	Typed intent objects, contexts, expectations and lifecycle handling.	Canonical typed objects and lifecycle logic are implemented; 3GPP management-service interface conformance is not tested.
Service refinement and closed-loop assurance	ETSI ZSM GS 009-1, GS 016, GS 018; ETSI ENI translation and conflict documents	Hierarchical closed loops, intent translation, digital-twin support and conflict handling.	Functions are represented in the local orchestration code; ETSI API or information-model conformance is not claimed.
RAN-domain actuation	O-RAN near-RT RIC, E2AP and E2 service-model concepts	Domain agents and bounded control actions correspond to RIC and near-real-time application (xApp) actuation points.	A bridge to ns-3 5G-LENA is supplied; the evaluated analytical plant and recovered bridge dry runs do not constitute E2AP wire-protocol conformance tests.
Analytics and state observability	3GPP TS 23.288, TS 28.104, TS 28.105	Analytics exposure and AI/ML lifecycle concepts support state estimation and assurance.	Equivalent telemetry and analytics variables are used in the simulation; NWDAF or management API conformance is not tested.
Security and AI-agent safeguards	3GPP TS 33.501, National Institute of Standards and Technology (NIST) AI Risk Management Framework (RMF) Generative AI (GenAI) Profile [31] and Open Worldwide Application Security Project (OWASP) GenAI guidance [32]	Telecom security is retained as the baseline while AI-risk governance, typed validation, output checking, least privilege and sandboxing address agent-specific risks.	Security mechanisms are design-level controls and simulated gates; certification or full conformance against these guidance documents is not claimed.

Abbreviations: application programming interface (API); E2 application protocol (E2AP); Network Data Analytics Function (NWDAF); near-real-time (near-RT).

2.2. Intent Modelling, Translation and Closed Loops

Several themes have converged in the literature on intent-based telecom and network automation. The broader intent-based networking lifecycle has been surveyed by Leivadeas and Falkner, where intent expression, translation, resolution, activation and assurance are treated as the main closed-loop functions of an autonomous intent-based networking system [33]. End-to-end intent-based networking for 5G and beyond is reviewed by Velasco et al., with emphasis on secure orchestration and the integration of analytics and service management beyond a northbound application programming interface (API) [34]. Hierarchical and peer interactions among intent-driven closed loops, including delegation and escalation across domains, are examined by Gomes et al. [35]. Decomposition and propagation of intents for network-slice design are studied by Gritli et al., through which the need for layered refinement is illustrated [36].

Mobile-specific work has made the hierarchy more concrete. Business, service and resource intents are proposed by Xie et al. for multi-vertical end-to-end network-slicing services and this hierarchy is adopted here [37]. Intent-driven orchestration is applied to cognitive autonomous RAN management by Banerjee et al., linking intent semantics to domain-specific control loops [38]. The integration of network-slice control into 5G/6G intent-based networking is studied by Da Costa and Contreras Murillo [39]. SLA management in intent-driven service management is systematised by Sharma et al. through a taxonomy of lifecycle activities [40].

Two recent directions are especially important. First, ETSI ENI has standardised multi-stage translation ideas, through which staged intent-to-policy transformation is formalised instead of leaving translation fully ad hoc [25]. Second, recent LLM-oriented work such as Network Meets ChatGPT and Confucius shows that natural-language ingress can be effective if it is constrained by schemas, tools and structured procedures [7,8]. A specialised 5G-core study has also shown how a custom LLM can classify standards-inspired intent types, including deployment, modification, performance assurance, reporting, feasibility checking and notification intents [41]. That observation matters for mobile networks because it suggests that agentic control should be bounded and tool-mediated rather than free-form, while also showing that intent extraction alone remains insufficient without distributed coordination, safety and trust control. A complementary 2026 direction is explanation-based runtime verification for ML-driven optical networks, where explanation coherence is checked together with physics-aware consistency before an ML decision is executed, and uncertain decisions may be deferred or rejected [42]. This is closely related to the safety objective pursued here, but the mechanisms are different: the present framework validates typed constraints, authorisation scopes and hard network invariants before actuation, while explanation-based verification can be added as an additional verifier when a learned component contributes a candidate action. It is thus treated as a complementary runtime-assurance mechanism rather than as an implemented component of the current prototype.

2.3. Distributed AI and Multi-Agent Control

A separate literature stream is focused on distributed optimisation and multi-agent control. Distributed handover management in dense millimetre-wave (mmWave) networks is studied using multi-agent reinforcement learning by Sana et al. [43]. Adaptive user association in dynamic mmWave networks is examined separately by the same research group [44]. These papers are not intent-centric, but they demonstrate an important point: locally observing agents can coordinate effectively in dense wireless systems without relying on a single omniscient optimiser. Agent-based distributed AI has also been applied directly to device-level control, where belief-desire-intention agents are used to govern

device-to-device communication in 5G through distributed agent coordination [6] and the broader space of AI/ML methods for device-to-device communication in 5G/6G and beyond has been surveyed in depth [45].

Graph-aware and slicing-aware papers further strengthen the case. Multi-agent graph convolutional reinforcement learning is used for intelligent load balancing by Houidi et al. [46]. Generalisable graph neural network (GNN) methods for slicing and admission control across 5G RAN/MEC metropolitan topologies are presented by Moayyedi et al. [47]. Multi-agent deep reinforcement learning is used for multi-objective RAN-slicing optimisation by Zhou et al. [48]. Graph convolution and reinforcement learning are combined for load balancing and smart queuing by Fawaz et al. [49]. Together, these works show that distributed and graph-structured decision models are increasingly natural for mobile deployments.

Even where learning is used, model-based control remains relevant. Chiputa et al. combine a jump Markov linear system with deep reinforcement learning for 5G mmWave handover and thereby demonstrate that local dynamics can be meaningfully structured rather than merely learned end to end [50]. That observation motivates the hybrid control-theoretic option developed later in this paper.

2.4. O-RAN Conflicts, Agentic Operations and Trust

The practical relevance of conflict management is particularly evident in Open RAN. PACIFISTA studies conflict evaluation and mitigation in O-RAN, showing that independently useful xApps may interact badly, and that conflict mitigation should be part of functional O-RAN stacks rather than an afterthought [51]. Recent O-RAN public-document releases also explicitly mention conflict mitigation and richer AI/ML interfaces in the RIC environment [30].

Meanwhile, agentic orchestration is entering network operations more broadly. Confucius shows that multi-agent LLM systems can structure network workflows as directed acyclic graphs, embed domain procedures and use tool access and retrieval to improve execution reliability [8]. Recent secure agentic intent-based networking work in the MARE project proposes a hierarchical multi-agent and multi-vendor architecture with LLM-supported intent processing, external security knowledge bases and a dedicated security-plane view [52]. In telecom-specific standardisation, ETSI ZSM and ENI have both moved into agent-oriented studies, indicating that agent-based approaches are becoming increasingly relevant to mobile-network management [22,27,28].

Finally, trust and privacy are receiving increasing attention in the DAI literature. Communication-efficient federated learning has been studied for O-RAN intelligent controllers [53]. Trustworthy reputation and blockchain-assisted accountability have been proposed for federated learning in O-RAN [54]. These works matter because intent-driven networks are likely to combine local data, distributed training and multi-vendor cooperation. A comparison of representative primary works and their remaining gaps is given in Table 3.

Table 3. Representative recent primary studies and their remaining research gaps.

Work	Year	Domain	Main Contribution	Remaining Gap
Leivadeas and Falkner [33]	2023	IBN lifecycle	Comprehensive intent expression, translation, activation and assurance taxonomy.	Survey-level treatment, not mobile DAI control.
Velasco et al. [34]	2021	End-to-end management	Strong 5G/6G intent-based networking vision.	Limited formal multi-agent theory.

Table 3. *Cont.*

Work	Year	Domain	Main Contribution	Remaining Gap
Gomes et al. [35]	2021	Closed loops	Hierarchical intent-driven closed loops.	No distributed convergence or stability proofs.
Gritli et al. [36]	2021	Slice design	Intent decomposition and propagation.	Limited runtime realisation detail.
Xie et al. [37]	2022	Multi-vertical slicing	Business/service/resource hierarchy.	No explicit optimisation framework.
Sana et al. [43,44]	2020	Mobility	Distributed multi-agent reinforcement learning (MARL) for handover and association.	No typed intent layer.
Houidi et al. [46]	2022	Load balancing	Graph-based multi-agent wireless control.	Not standards-aligned with telecom intents.
Moayyedi et al. [47]	2023	Slicing and admission	GNN-based generalisation across topologies.	Intent lifecycle and trust not central.
PACIFISTA [51]	2025	O-RAN	Practical O-RAN conflict mitigation.	Does not solve end-to-end refinement.
Habibi et al. [3]	2024	O-RAN service management and orchestration (SMO)	AI/ML-driven SMO scenarios and challenges.	Strong architecture, limited formal proofs.
Manias et al. [41]	2024	5G core intent ingress	LLM-based extraction of 3GPP-inspired 5G core intent classes.	Ingress only, without distributed execution.
Confucius [8]	2025	Network operations	Multi-agent LLM operational workflow framework.	Mobile-network control guarantees remain open.
Zacarias et al. [52]	2026	Secure agentic IBN	Multi-agent, multi-vendor intent-based security architecture with LLM support.	Security-centric and not evaluated as mobile DAI control.

Abbreviations: distributed artificial intelligence (DAI); intent-based networking (IBN); large language model (LLM); graph neural network (GNN); sixth-generation (6G).

The scope of the principal method families is compared against the seven criteria in Table 4.

Table 4. Criteria-based comparison of the related families with this work.

Family	Intent Typing	Distributed Optimisation	Convergence Guarantee	Trust	Security	O-RAN Integration	Evaluation Evidence
3GPP/ETSI closed-loop intent frameworks [5,18,30]	yes	no	no	partial	partial	partial	no (specification)
Intent hierarchies and decomposition [36,37]	yes	partial	no	no	no	partial	partial (simulation)
LLM intent ingress [7,41]	partial	no	no	no	no	no	partial
Distributed MARL mobility control [43,44]	no	yes	partial	no	no	no	yes (simulation)
Graph-based RAN and slicing control [47,48]	no	yes	partial	no	no	partial	yes (simulation)
O-RAN conflict mitigation [51]	no	no	no	no	no	yes	yes (testbed)
Tool-mediated multi-agent workflows [8]	partial	partial	no	partial	partial	no	yes
Secure agentic IBN [52]	yes	partial	no	yes	yes	no	partial
This work	yes	yes	yes (instantiated with attribution)	yes	yes	mapped to 3GPP, ETSI and O-RAN interfaces	analytical plant (twenty seeds, twelve controllers); bridge dry run; coordination benchmark (4 to 64 domains)

2.5. Research Gap and Position of This Work

The recent literature contains valuable but individually incomplete contributions. Intent-oriented papers define hierarchies [34,35], decomposition [36] or architecture [37,39], yet runtime guarantees such as feasibility preservation, convergence and stability are rarely stated formally. Multi-agent RAN papers show how distributed learning or graph-based optimisation can control load balancing [46,49], association and handover [43,44] and slicing [47,48], but they are seldom connected to typed intent semantics or to telecom standardisation. O-RAN papers and standards address multi-app conflicts and open interfaces [30,51] without supplying an end-to-end refinement theory. Standards documents establish a precise vocabulary for intent objects, closed loops and analytics exposure, but a vocabulary is not an algorithm and the question of how a high-level expectation is realised under coupled resource limits is left open. Recent agentic-systems papers show that bounded, tool-mediated orchestration is effective [8], although operator-grade guarantees are not their primary focus.

Table 4 locates each family against seven criteria. The comparison motivates a standards-grounded treatment that unifies (a) formal intent objects and hierarchies, (b) distributed mobile-network control, (c) the optimisation and control derivations, (d) the properties they carry, namely feasibility, convergence and stability, (e) security, privacy and trust for agentic control and (f) a reproducible evaluation. The element that is missing across these strands is a coordination mechanism for coupled resources that operates under administrative separation: intent-oriented and O-RAN treatments allocate from aggregate declarations or assume a coordinator with full information, whereas multi-agent learning treatments coordinate implicitly, without a feasibility certificate or an admission rule for the exchanged messages. What is proposed here is therefore centred on shared-space coordination: typed intent objects are connected to a constrained optimisation problem whose coupling is expressed in a common resource space, joint feasibility is certified before economic coordination and trust admission is applied to the exchanged contributions. Lyapunov-guided execution, finite potential-game conflict resolution and the mapping onto standardised actuation points are retained as the supporting stack and rest on known results. This focus also fixes the scope of the evaluation: the mechanism is assessed in a coordination-layer benchmark in which local models cannot be shared and its embedding in the complete stack is supported by a recovered bridge dry run and by an in-depth seeded system-level campaign on an analytical plant, where the score of the best comparator is matched by the complete controller, which is the only configuration in which trust admission, certified feasibility and bounded controller state are combined; a reproducible path to packet-level validation is supplied. In Table 4, “yes” means that the criterion is a first-class element of the cited work, “partial” that it is present in a restricted form and “no” that it is absent; the last row states the status of this work, including what is only mapped or only instantiated.

3. System Model, Formal Definitions, Notation and Problem Formulation

The formal methodology used to transform a typed network intent into a jointly feasible and safely executable distributed control action is developed in this section. The architecture, its timescales and the formal entities of agents, intents, network state and mobility state are defined first, together with the BDIX agent cycle and the assumptions under which the later guarantees hold. The refinement of business intents into service and resource children is then formalised with the notation and the intent-fulfilment objective and the construction and verification of sound refinement operators is described. The constraint families of resources, latency, reliability, mobility and trust are collected and the Phase-I minimum-slack certificate by which joint feasibility is separated from performance

optimisation is established, followed by the shared-space coordination problem with its convergence and stability properties. Finally, discrete conflict resolution through a finite potential game is connected to local safe execution and the interpretation of the resulting guarantees is stated.

The methodology is constructed as a sequence of contracts rather than as a single opaque optimiser. A high-level intent is first converted into typed predicates and context selectors. Those objects are refined into service and resource children together with explicit coupling information. Joint feasibility is then checked before performance optimisation is allowed to proceed. When the minimum required relaxation is zero, the operating optimisation is solved in a shared coordination space so that agents with heterogeneous local variables can still coordinate through semantically compatible quantities. The resulting domain budgets are passed to fast local executors, where Lyapunov weights and hard safety invariants determine whether an action can be applied directly, must be modified or must be replaced by a safe fallback. Discrete policy conflicts are handled separately through the finite potential-game mechanism so that continuous resource coordination and discrete template selection are not conflated.

Each stage has a distinct failure status. An invalid or unauthorised request is rejected during intent typing. Positive Phase-I slack indicates that the represented child-resource constraints cannot be satisfied jointly without relaxation. Failure to meet the coordination stopping criterion is reported separately from resource feasibility. If a proposed action violates a local invariant, it is modified or replaced by the safety filter. These statuses are retained separately in the implementation trace so that the reason for a deviation from the requested intent can be identified.

3.1. Architecture, Timescales and Formal Definitions

A managed mobile network is represented by the time-varying graph of Equation (1)

$$\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t), \tag{1}$$

where the node set of Equation (2),

$$\mathcal{V} = \mathcal{V}_{\mathrm{RAN}} \cup \mathcal{V}_{\mathrm{tr}} \cup \mathcal{V}_{\mathrm{MEC}} \cup \mathcal{V}_{\mathrm{CN}} \tag{2}$$

covers radio, transport, edge/cloud and core entities. The edge set is time varying because user associations, wireless links and some logical service chains change over time. The agent fabric is itself the graph of Equation (3)

$$\mathcal{G}^{\mathrm{ag}} = (\mathcal{A}, \mathcal{E}^{\mathrm{ag}}), \tag{3}$$

whose nodes are management and execution agents and whose edges denote trusted coordination channels.

Four layers are used by the architecture. The business layer performs operator- or tenant-facing intent ingestion. The service layer performs service and slice refinement, negotiation and assurance. The domain layer hosts RAN, transport, MEC and core domain agents. The execution layer performs local actuation through near-real-time applications (xApps), schedulers, policy engines and safe fallbacks. The proposed architecture is illustrated in Figure 1. Four functional control layers are represented across five visual bands. The top band contains business ingress and the Service Intent Agent, which belongs functionally to the service layer; service functions continue in the service management and orchestration (SMO) and intent-orchestration band. The domain-agent and execution bands represent the domain and execution layers. The managed-infrastructure band is

the controlled plant. Thus, the visual boundary between business ingress and service refinement does not define an additional control layer. Typed intents are refined and delegated, authorised actions are enacted and telemetry is returned for assurance and what-if evaluation. In the figure, Management Data Analytics Function (MDAF), O-RAN distributed unit (O-DU), O-RAN central unit (O-CU), next-generation Node B (gNB), user equipment (UE), non-real-time (Non-RT) operation and non-real-time applications (rApps) are represented.

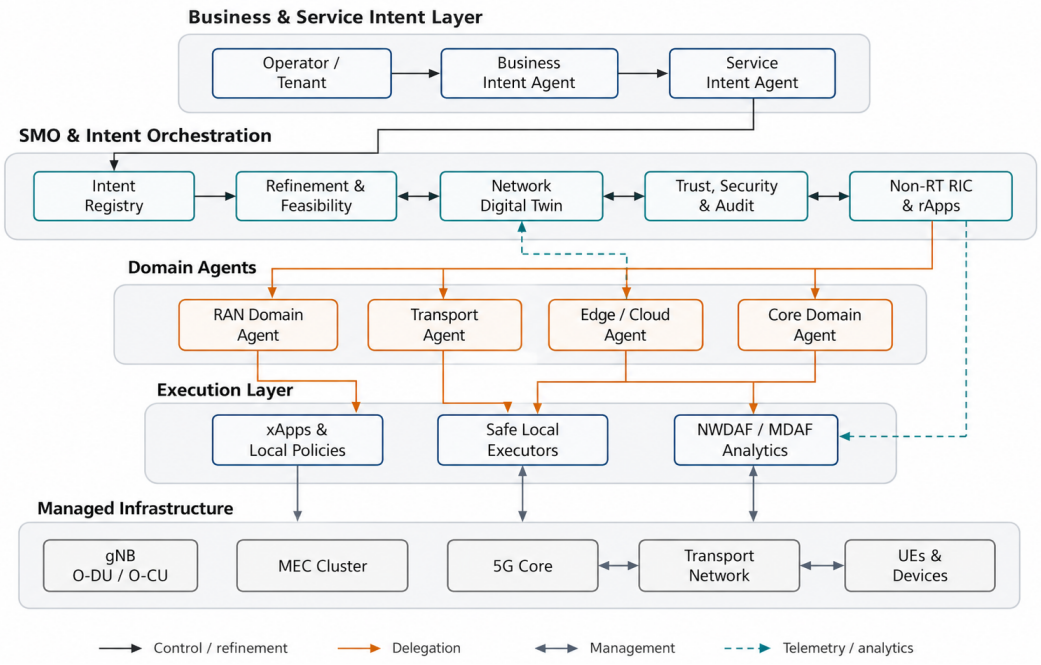


Figure 1. Proposed standards-aligned DAI architecture for agentic intent-based networking. Abbreviations: distributed artificial intelligence (DAI); service management and orchestration (SMO); radio access network (RAN); RAN intelligent controller (RIC); non-real-time (Non-RT); near-real-time applications (xApps); non-real-time applications (rApps); Network Data Analytics Function (NWDAF); Management Data Analytics Function (MDAF); next-generation Node B (gNB); O-RAN distributed unit (O-DU); O-RAN central unit (O-CU); multi-access edge computing (MEC); fifth-generation (5G); user equipment (UE).

The architecture operates across the control timescales of Equation (4):

$$\Delta_{\text{biz}} \gg \Delta_{\text{svc}} \gg \Delta_{\text{dom}} \gg \Delta_{\text{exec}}. \tag{4}$$

Business-intent reassessment is therefore slowest, execution is fastest and domain coordination lies in between. This mirrors the practical split between non-real-time orchestration, service assurance and near-real-time RAN control [1,30].

The agent, intent, hierarchy, network state and mobility model are specified in Definitions 1, 3, 4, 5 and 6, respectively.

Definition 1 (Agent). *An agent $a_i \in \mathcal{A}$ is the tuple of Equation (5),*

$$a_i = (\Omega_i, \mathrm{Y}_i, \Pi_i, \Sigma_i, \Theta_i), \tag{5}$$

where Ω_i is its observation space, Y_i its action space, Π_i its decision-policy family, Σ_i its authorised interfaces and tools and Θ_i its trust and safety envelope.

Definition 2 (BDIX agent). *A BDIX agent is an agent of Definition 1 whose policy is realised by the cycle* perceive → deliberate → commit → execute. *Its belief state contains the typed observation, the digital-twin estimates, the capped controller deficits q_m and peer reputations. Perception updates observations, the twin is fitted from realised outcomes and the deficit recurrence specified below is applied once per epoch. Candidate desires comprise refined resource budgets, coordinated allocations and discrete templates. Commitment selects policy-admissible candidates; execution then applies the safety projection and the final resource check. A successful resource check certifies only the represented box and pool constraints. A failed predictive KPI check is retained as an explicit degraded-service warning after the bounded negotiation budget; it is not silently converted into a fulfilment certificate. If no intersection exists between the current surrogate box and the hard safety envelope, only the hard resource invariants are enforced and* `resource_certified` *is false. Trust admission governs peer proposals, while resource-slice agents remain distinct from the advisory peers. These implementation choices instantiate the four BDIX stages but do not prove semantic parent-intent satisfaction. Table 5 locates the stages in the architectural algorithms.*

Definition 3 (Intent). *An intent $\iota \in \mathcal{I}$ is the declarative object of Equation (6),*

$$\iota = (\kappa, \tau, \mathcal{C}, \omega, \rho, \eta), \tag{6}$$

where κ is the set of desired KPI or outcome predicates, τ is the temporal scope, $\mathcal{C}$ is the context set, ω is a priority vector, ρ is refinement metadata and η contains security and trust requirements.

Table 5. Belief-Desire-Intention-eXecution (BDIX) stages and where they appear in Algorithms 1–3.

Stage	State Touched	Algorithm Lines
Perceive	typed state estimate, twin model, virtual queues, trust scores	Algorithm 1 normalisation and validation; Algorithm 2 message reception, hard checks and trust evidence update; Algorithm 3 observation and queue update.
Deliberate	candidate child intents, candidate allocation, candidate templates	Algorithm 1 refinement $R_{B\to S}$, $R_{S\to R}$ and Phase-I solve; Algorithm 2 proximal and extrapolation steps; Algorithm 3 template assembly or optional receding-horizon solve.
Commit	policy-admissible candidate; final resource status and unresolved KPI warning	Algorithm 1 feasible-branch return or renegotiation; Algorithm 2 admission rule and aggregate recovery; Algorithm 3 hysteresis and safety decision.
Execute	governed actuation through authorised tools	Algorithm 1 budget attachment; Algorithm 2 returned allocation; Algorithm 3 enactment, fallback and logging.

Definition 4 (Intent hierarchy). *The intent hierarchy is a partially ordered set $(\mathcal{L}, \preceq)$ with $\mathcal{L} = \{B, S, R\}$ and $B \prec S \prec R$, where B, S and R denote business, service and resource levels, respectively, and $\prec$ denotes increasing operational specificity. A refinement operator $R_{\ell\to\ell'}$ maps a parent intent and a state estimate into a set of child intents and associated coupling constraints.*

Definition 5 (Network state). *The global state at time t is given by Equation (7),*

$$s_t = (q_t, \ell_t, c_t, m_t, z_t), \tag{7}$$

where queue and load variables are collected in q_t, channel or link variables in ℓ_t, compute variables in c_t, mobility variables in m_t and security, trust and model-health variables in z_t.

Definition 6 (Mobility model). *For user $u \in \mathcal{U}$, the mobility state is given by Equation (8),*

$$\xi_u(t) = (p_u(t), v_u(t), \chi_u(t)), \tag{8}$$

where $p_u(t)$ is position, $v_u(t)$ is velocity and $\chi_u(t)$ is a contextual mode such as pedestrian, vehicular or stationary. The model may be analytical, trace-driven or hybrid.

Typed control, bounded actions, a convex surrogate, connectivity and a safety override are required by Assumptions 1, 2, 3, 4 and 5, respectively.

Assumption 1 (Typed internal representation)**.** *Free-form natural language may be accepted at ingress, but all optimisation and actuation operate on typed internal intent objects.*

Assumption 2 (Bounded action sets)**.** *Each domain uses either a compact convex decision set $X_i \subset \mathbb{R}^{d_i}$ or a finite policy-template set P_i.*

Assumption 3 (Convex slow-timescale surrogate)**.** *At the service and resource levels, the realisation problem admits a convex surrogate with separable local objectives and affine coupling constraints.*

Assumption 3 is a relaxation and what it relaxes is stated here explicitly. Three elements of the true realisation problem are not convex. User association is integer; in the framework it is not a slow-timescale decision variable but is left to the execution layer, which steers it through the association bias and the handover hysteresis of the mobility-aware safe execution (Section 4.4), so the surrogate never rounds an association. Scheduler-template selection is discrete; it is removed from the convex problem and handled by the finite potential game of Section 3.5, so the surrogate decides only the continuous pooled shares. Interference coupling through the signal-to-interference-plus-noise ratio (SINR) is non-convex; the per-slice response coefficients are fixed during each coordination solve (radio capacity per unit share and latency coefficients for the compute and transport shares), which makes the per-epoch objective separable and convex at the cost of a modelling error that is re-estimated every epoch. The approximation error of the full plant is not bounded here. Section 6.5.1 distinguishes historical best-found plant searches from a certified finite-profile benchmark whose scope is explicitly restricted.

Assumption 4 (Connected coordination graph)**.** *During a coordination epoch, the agent graph is connected and only authenticated messages are admitted.*

Assumption 5 (Safety override)**.** *Every execution agent has access to a local safety override that can reject an otherwise admissible action if a hard invariant would be violated.*

3.2. Intent Refinement, Notation and Objective

Business intents are adopted for operator or tenant outcomes, service intents for slice and service budgets and assurance targets and resource intents for domain-specific executable policies and budgets. A refinement chain is expressed as in Equation (9),

$$\iota^B \xrightarrow{R_{B\to S}} \{\iota_j^S\}_{j=1}^{n_S} \xrightarrow{R_{S\to R}} \{\iota_{jk}^R\}_{k=1}^{n_{R,j}}. \tag{9}$$

The soundness of the operators in Equation (9) is a property that has to be constructed and verified rather than assumed; the finite composition checks for declared threshold semantics and the separate Phase-I resource-feasibility certificates are described in Section 3.3.

Definition 7 (Sound refinement)**.** *A refinement operator is* sound *if every* joint *child realisation that fulfils all child predicates and all refinement-generated coupling constraints also fulfils the parent predicate. Soundness is therefore an implication about fulfilment and does not assert that the child set is feasible.*

For a business intent ι^B, let $\mathcal{F}_R(\iota^B)$ denote the joint resource-level feasible set containing all stacked resource decisions whose local domain constraints, child-intent constraints and refinement-generated coupling constraints are satisfied simultaneously. In particular, feasibility of every child considered separately is not sufficient to conclude that $\mathcal{F}_R(\iota^B)$ is non-empty.

Proposition 1 (Fulfilment preservation by composition of sound refinements)**.** *If $R_{B\to S}$ and $R_{S\to R}$ are sound, $\mathcal{F}_R(\iota^B) \neq \emptyset$ and a realisation $x \in \mathcal{F}_R(\iota^B)$ fulfils all generated resource-level child predicates, then the corresponding service-level predicates and the originating business-intent predicate are fulfilled. Consequently, the parent intent is feasible whenever such a jointly feasible child realisation exists. The statement is the composition of the two soundness implications of Definition 7 and is recorded because the framework relies on it; the substantive requirement is the construction of sound operators, which is addressed in Section 3.3.*

Proof. Because $x \in \mathcal{F}_R(\iota^B)$, the complete set of local child constraints and refinement-generated couplings is satisfied jointly. Soundness of $R_{S\to R}$ therefore maps fulfilment of the resource-level children to fulfilment of their service-level parents. Soundness of $R_{B\to S}$ then maps fulfilment of the service-level children to fulfilment of ι^B. The conclusion follows by composition of the two sound refinement implications. The joint-feasibility premise is essential because independently feasible children may still conflict through shared capacities or refinement-generated couplings. □

The main symbols used in the formulation are listed in Table 6.

Table 6. Main notation.

Symbol	Meaning	Symbol	Meaning
$\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t)$	Time-varying managed mobile-network graph.	$u_i(t)$	Fast-timescale actuation of execution agent i.
$\mathcal{G}^{\mathrm{ag}} = (\mathcal{A}, \mathcal{E}^{\mathrm{ag}})$	Agent communication graph.	$G_i x_i \in \mathbb{R}^p$	Contribution of agent i to the common p-dimensional coupling space.
$\iota \in \mathcal{I}$	Intent object.	b	Global capacity or policy budget vector.
κ	KPI or outcome predicates in an intent.	λ	Dual vector for coupled constraints.
$\mathcal{C}$	Context set attached to an intent.	ζ	Non-negative Phase-I slack in the common coupling space.
ω	Priority vector.	$Q_m(t)$	Virtual queue for violation process m.
s_t	Global network state at time t.	$L(Q)$	Lyapunov function.
$\hat{s}_{i,t}$	Local state estimate of agent i.	V	Drift-plus-penalty trade-off parameter.
X_i^{ι}	Local convex set including agent i constraints induced by the delegated intent.	$\xi_u(t)$	Mobility state of user u.
P_i	Discrete policy-template set of agent i.	$w_{ij}(t)$	Trust score assigned by agent i to agent j.
$x_i \in \mathbb{R}^{d_i}$	Slow-timescale decision vector of domain agent i.	$\Phi(a)$	Potential function in the conflict game.

Let $y_m(t)$ denote KPI $m \in \mathcal{M}$ at time t and let $v_m(t;\iota)$ denote its non-negative violation magnitude under intent ι. For a latency target $D_s(t) \leq D_s^{\max}$, for example, Equation (10) gives

$$v_m(t;\iota) = \max\{D_s(t) - D_s^{\max}, 0\}. \tag{10}$$

The instantaneous utility is defined in Equation (11) as

$$\begin{aligned} U_t = & \sum_{m\in\mathcal{M}} \omega_m u_m(y_m(t),\iota) \\ & - \alpha_E \tilde{E}(t) - \alpha_H \tilde{H}(t) - \alpha_C \tilde{C}_{\mathrm{chg}}(t) - \alpha_S \tilde{R}_{\mathrm{sec}}(t), \end{aligned} \tag{11}$$

where $\tilde{E}(t)$ is the normalised energy, $\tilde{H}(t)$ is the normalised mobility overhead, $\tilde{C}_{\mathrm{chg}}(t)$ is the normalised configuration churn and $\tilde{R}_{\mathrm{sec}}(t)$ is the normalised security-risk penalty; the tilde denotes a dimensionless quantity scaled to $[0,1]$ as specified next, so that every term of Equation (11) is dimensionless and the equation matches the notation of the score in Equation (46). Each physical quantity is normalised before combination. Throughput, latency and reliability deficits are divided by their respective positive targets and clipped to $[0,1]$, giving dimensionless fulfilment $1-\hat{v}_m$, where $\hat{v}_m = \min\{v_m/v_m^{\max}, 1\}$ and $v_m^{\max}$ is the corresponding positive target scale. The priority of each slice is assigned to each of its three predicates and all predicate weights are normalised jointly. Security is represented by a separate risk penalty in the analytical endpoint. The implemented energy reference is $E^{\mathrm{ref}} = (1.4 N_{\mathrm{cell}} P_{\mathrm{tx}} + 35)\Delta t$ joules; this includes the configured maximum transmit-power multiplier and a 35 W reference overhead. The normalised handover rate, mean absolute pooled-share change and mean security-risk score are clipped to $[0,1]$. The configured cost weights $(\alpha_E, \alpha_H, \alpha_C, \alpha_S) = (0.08, 0.05, 0.05, 0.08)$ therefore bound the cost penalty by 0.26. These values are shared by all controllers and are fixed before evaluation. Weight sensitivity is treated as exploratory rather than as evidence that all possible weight choices preserve a ranking. The long-run constrained problem is

$$\max_{\pi} \quad J(\pi) \triangleq \liminf_{T\to\infty} \frac{1}{T}\mathbb{E}_{\pi}\left[\sum_{t=0}^{T-1} U_t\right] \tag{12}$$

$$\text{s.t.} \quad \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{\pi}[g_r(t)] \le 0, \quad \forall r \in \mathcal{R}. \tag{13}$$

The objective in Equation (12) is evaluated subject to the average constraints in Equation (13), where π is a control policy and $g_r(t)$ is the signed residual of constraint r, with positive values indicating a violation.

3.3. Constructing and Verifying Sound Refinement Operators

Proposition 1 requires two separate properties: satisfaction of the generated child predicates must imply the parent predicate and the complete child set must have a joint realisation. Neither property follows from checking children individually. For a declared serial-path template, a sufficient implication check is obtained by summing latency upper bounds, taking the minimum of throughput lower bounds and composing reliability bounds under an explicitly stated dependence model. The executable helper `verify_composition` checks these finite threshold conditions. Independent segments, or correctly specified conditional reliability bounds, permit a product rule; without independence a conservative union bound is used. The helper rejects missing, invalid or non-finite bounds. This is a check of declared composition semantics, not a verification of the learned resource-to-KPI map.

A parent requiring end-to-end reliability of at least 0.97 provides a counterexample to naive decomposition. Two independent segments can each satisfy a child target of 0.97 while their joint reliability is only $0.97^2 \approx 0.94$. Assigning both children $\sqrt{0.97}$ makes the product at least the parent target under the independence assumption. Without that assumption, the union-bound construction requires $1 - \sum_j (1 - r_j) \ge 0.97$, for example $r_1 = r_2 = 197/200$, or 98.50%. Automated regression cases accept the valid construction and reject the naive one. Additive latency bounds are sound only when every delay contribution is counted once; shared queues or overlapping service segments must be represented explicitly.

The supplied controller uses heuristic resource refinement and per-epoch Phase-I checks. Its learned KPI estimates are not interval enclosures and no automatic proof of their monotonicity or parent implication is claimed. Phase I certifies non-emptiness only for the represented convex resource surrogate. A positive minimum slack establishes infeasibility of that child surrogate, not impossibility of every implementation of the business intent. Consequently, the composition proposition remains conditional for the full stochastic plant. The separate composition examples, resource checks and realised KPI logs make clear which properties are mechanically checked and which remain modelling assumptions.

3.4. Constraint Families and Phase-I Joint Feasibility

Typical constraints include the following. For cell n, $b_{u,s,n}$ is the radio allocation to user u in slice s, $B_n^{\max}$ is the radio budget, p_n is transmit power and $P_n^{\max}$ is its limit. Compute allocation $c_{s,v}$ and transport flow $f_{s,e}$ are bounded by node capacity $C_v^{\max}$ and link capacity $F_e^{\max}$. The long-run mean energy $\overline{E}$ is bounded by $\bar{E}^{\max}$. Slice latency D_s and reliability R_s have targets $D_s^{\max}$ and $R_s^{\min}$, with admissible outage probabilities ϵ_s and δ_s. The handover budget $H_u^{\max}(W)$ applies over a window of W slots.

Radio capacity is constrained by Equation (14),

$$\sum_{s\in\mathcal{S}}\sum_{u\in\mathcal{U}} b_{u,s,n}(t) \le B_n^{\max}. \tag{14}$$

Power and energy are constrained by Equation (15),

$$0 \le p_n(t) \le P_n^{\max}, \qquad \overline{E} \le \bar{E}^{\max}. \tag{15}$$

Edge and cloud compute resources satisfy Equation (16),

$$\sum_{s\in\mathcal{S}} c_{s,v}(t) \le C_v^{\max}. \tag{16}$$

Transport capacity satisfies Equation (17),

$$\sum_{s\in\mathcal{S}} f_{s,e}(t) \le F_e^{\max}. \tag{17}$$

Latency and reliability are captured through Equation (18),

$$\begin{aligned} \mathbb{P}(D_s(t) \le D_s^{\max}) &\ge 1-\epsilon_s, \\ \mathbb{P}(R_s(t) \ge R_s^{\min}) &\ge 1-\delta_s. \end{aligned} \tag{18}$$

Mobility stability is imposed as in Equation (19),

$$\sum_{\tau=t}^{t+W-1} h_u(\tau) \le H_u^{\max}(W), \tag{19}$$

where $h_u(\tau) \in \{0,1\}$ indicates a handover.

A control proposal from agent j is admissible to agent i only under Equation (20),

$$w_{ij}(t) \ge \underline{\theta}_{ij}. \tag{20}$$

Here $w_{ij}(t)$ is the trust score that is current at the decision epoch; the update order of the score relative to the admission decision is specified in Section 5.3.

Joint feasibility is checked before economic coordination through a Phase-I minimum-slack problem rather than by interpreting a finite penalty inside the original objective:

$$\rho^\star = \min_{\{x_i\},\zeta\succeq 0} \quad \mathbf{1}^\mathsf{T}\zeta \tag{21}$$

$$\text{s.t.} \quad \sum_{i\in\mathcal{A}^D} G_i x_i \le b + \zeta, \tag{22}$$

$$x_i \in X_i^\iota, \quad \forall i. \tag{23}$$

The minimum-slack objective in Equation (21), the shared coupling constraint in Equation (22) and the local feasible sets in Equation (23) together define the Phase-I joint-feasibility certificate.

The index set $\mathcal{A}^D$ contains the domain agents and the sets X_i^ι include their delegated resource-level requirements. Assume that these sets are nonempty and compact, so the continuous Phase-I objective attains its minimum. An empty local set must instead be reported as local infeasibility before this finite-slack program is applied. Under these conditions, $\rho^\star = 0$ if and only if the current convex surrogate admits a zero-slack joint realisation. If $\rho^\star > 0$, the surrogate is jointly infeasible and the support and magnitude of an optimal $\zeta^\star$ identify an ℓ_1-minimum relaxation of the represented coupling constraints. This Phase-I construction is deliberately separated from the operating objective, so a non-zero slack cannot be caused merely by a finite trade-off between relaxation and performance.

Two typed-floor parameters are used in the selected design variant of Section 6.4. First, physical backlog is incorporated into the radio-demand estimate as $\tilde{D}_i = \bar{D}_i + g B_i^{\mathrm{phys}}$, where $\bar{D}_i$ is smoothed offered load in Mbps, B_i^{phys} is the preceding physical backlog in Mbit and $g = 1.25\,\mathrm{s}^{-1}$. This term raises the represented demand when traffic remains queued; it does not guarantee that the backlog is drained within one epoch. Physical backlog is distinct from the capped deficit state q_m defined below. Second, a latency-margin uplift of 0.10 is applied to slices whose latency target is at most 1.20 times the tightest target, with the resulting margin capped at 0.60. This raises their represented compute and transport floors. Both changes preserve the form of the convex surrogate, while its feasibility and convergence assumptions must still be checked for the resulting parameters.

The certificate is operated with the declared numerical tolerance. Positive $\rho^\star$ identifies oversubscription of the represented child-resource set. Its support locates the affected pools and its magnitude is the minimum coupling slack for that surrogate. The campaign uses the headroom revision policy: available headroom and recent latency urgency determine which child floors are reduced first. Proportional and inverse-priority policies are supported alternatives but are not the campaign setting. Floors remain above the numerical minimum of 10^{-3}, so successful repair is verified by solving Phase I again rather than assumed. The post-revision value is exported separately. Priorities weight the operating objective and the chosen revision policy determines their role in floor relaxation.

An additional check is applied to the final action after template selection and the safety filter. Each resource vector is projected onto the intersection of its current lower and upper bounds with the unit simplex; this prevents normalisation from destroying a lower-bound certificate. If that intersection is empty, the hard safety envelope is enforced, and the action is explicitly reported as uncertified for the surrogate. Solver convergence, original Phase-I slack, post-revision slack, final resource certification and unresolved twin predictions are separate trace fields. A finite-iteration candidate is not called an optimum solely because a residual is small. The preserved guarantee is the feasibility of the checked resource intersection when certification succeeds. Neither a positive original slack nor a successful relaxed certificate proves or disproves the existence of another realisation of

the original parent intent. Operational acceptance of a degraded intent remains a policy decision; the simulator applies the configured revision policy automatically.

After a zero-slack joint realisation has been certified, slow-timescale coordination solves

$$\min_{\{x_i\}} \quad \sum_{i \in \mathcal{A}^D} f_i(x_i) \tag{24}$$

$$\text{s.t.} \quad \sum_{i \in \mathcal{A}^D} G_i x_i \leq b, \tag{25}$$

$$x_i \in X_i^{\iota}, \quad \forall i. \tag{26}$$

The operating objective in Equation (24) is solved under the shared coupling requirement in Equation (25) and the local intent-feasible sets in Equation (26).

The local objective used by the analytical implementation is made explicit in Equation (27):

$$f_i(x_i) = \sum_{j=1}^{d_i} \left[\frac{a_{ij}}{x_{ij}} + c_{ij} x_{ij} + \frac{\gamma_{ij}}{2} (x_{ij} - x_{ij}^{\text{ref}})^2 \right]. \tag{27}$$

Here $a_{ij} \geq 0$ weights an inverse-share penalty, c_{ij} is a signed linear coefficient, $\gamma_{ij} > 0$ is a proximity coefficient and x_{ij}^{ref} is the preceding allocation. The term a_{ij}/x_{ij} is omitted when $a_{ij} = 0$; a strictly positive lower bound is imposed whenever $a_{ij} > 0$. The coefficients and bounds are held fixed during one solve. Bounded deficits and advisory factors contribute to these coefficients. This per-epoch convex surrogate is not identical to direct optimisation of the stochastic objective in Equation (12).

Heterogeneous local dimensions d_i are permitted because consensus is not taken directly between x_i and x_j. Instead, each local decision is mapped by G_i into the shared p-dimensional coordination space. Let $\mathsf{K} = [G_1 \ \cdots \ G_N]$ denote the linear coupling operator for the stacked variable. A primal-dual hybrid-gradient (PDHG) update [9] is given by Equations (28)–(30), where I_X is zero on X and $+\infty$ outside it, prox denotes the proximal operator and $[\cdot]_+$ projects onto the non-negative orthant.

$$x_i^{k+1} = \text{prox}_{\tau(f_i + I_{X_i^{\iota}})}\left(x_i^k - \tau G_i^{\mathsf{T}} \lambda^k \right), \tag{28}$$

$$\bar{x}_i^{k+1} = 2x_i^{k+1} - x_i^k, \tag{29}$$

$$\lambda^{k+1} = \left[\lambda^k + \sigma \left(\sum_i G_i \bar{x}_i^{k+1} - b \right) \right]_+. \tag{30}$$

The extrapolated primal state in Equation (29) is used in the dual update. Thus, the exchanged coordination objects can be resource contributions $G_i x_i \in \mathbb{R}^p$ and the common dual vector $\lambda \in \mathbb{R}^p$, which remain dimensionally compatible even when $d_i \neq d_j$. There is no missing slack iterate in this operating-stage update because slack is confined to the preceding Phase-I feasibility check.

Proposition 2 (Convergence of the shared-space coordination; instantiation of the primal-dual hybrid-gradient result of Chambolle and Pock [9]). *Let $F(x) = \sum_i \left(f_i(x_i) + I_{X_i^{\iota}}(x_i) \right)$ be proper, closed and convex and assume that the zero-slack coordination problem Equations (24)–(26) admits a saddle point, for example under a suitable Slater condition. If the fixed step sizes satisfy $\tau\sigma\|\mathsf{K}\|_2^2 < 1$, then the iterates in Equations (28)–(30) converge to a primal-dual solution of the coordination problem. If F is strongly convex, the primal optimum is unique; uniqueness of the dual multiplier or of the saddle point is not asserted without additional assumptions.*

Proof. The problem has the standard convex saddle form $\min_x F(x) + I_{\{z \preceq b\}}(\mathsf{K}x)$. Under the stated existence condition and $\tau\sigma\|\mathsf{K}\|_2^2 < 1$, the iterates converge to a saddle point by the standard finite-dimensional PDHG convergence result [9]. Strong convexity of F makes the primal minimiser unique, while dual uniqueness requires separate regularity and is thus not used. □

The online stability component is then expressed through Lyapunov control. For each protected intent predicate m, the virtual queue is defined as

$$Q_m(t+1) = \max\{Q_m(t) + v_m(t;\iota) - \bar{v}_m, 0\}, \tag{31}$$

where $\bar{v}_m$ is the long-term admissible violation budget. With the Lyapunov function of Equation (32),

$$L(Q(t)) = \frac{1}{2}\sum_{m\in\mathcal{M}} Q_m(t)^2, \tag{32}$$

the conditional drift is $\Delta(t) = \mathbb{E}[L(Q(t+1)) - L(Q(t)) \mid Q(t)]$. A standard quadratic expansion yields

$$\begin{aligned}\Delta(t) - V\mathbb{E}[U_t \mid Q(t)] \le B \\ + \sum_{m\in\mathcal{M}} Q_m(t)\mathbb{E}[v_m(t;\iota) - \bar{v}_m \mid Q(t)] \\ - V\mathbb{E}[U_t \mid Q(t)]\end{aligned} \tag{33}$$

for some finite constant B when violations are bounded.

Proposition 3 (Queue stability and $O(1/V)$ performance; the drift-plus-penalty bound of Neely [10] for the virtual queues of Equation (31))**.** *Assume finite initial queues, bounded instantaneous utility and violation increments and independent and identically distributed exogenous states. Assume also that a stationary randomised policy based on the exogenous state satisfies the constraints with Slater slack and that stationary policies attain or approach $U^\star$. These conditions permit the queue-conditional comparison used in the drift bound. If each slot minimises the right-hand side of Equation* (33)*, then all virtual queues are mean-rate stable and the achieved time-average utility satisfies Equation* (34)*,*

$$\overline{U} \ge U^\star - \frac{B}{V}, \tag{34}$$

where $U^\star$ is the optimal utility of the constrained stochastic problem.

The bound applies to the uncapped queues of Equation (31) and exact slotwise minimisation under the stated stochastic assumptions. The correlated mobility process and learned controller used in the experiments are not certified to satisfy those assumptions. With a uniformly bounded additive optimisation error, the corresponding error term must be added to the numerator. It does not automatically apply to an H-step plan, a learned surrogate or a clipped controller state. In particular, the implemented state is

$$q_m(t+1) = \min\{8, \max[0, q_m(t) + d_m(t) - 0.02]\}, \tag{35}$$

where d_m is the signed target-relative deficit (security uses risk minus its allowable level). Negative deficits credit overfulfilment. This differs from the non-negative violation queue in Equation (31); bounded q_m follows from clipping and establishes neither mean-rate stability of the ideal queue nor stability of the physical traffic queues. An uncapped diagnostic $Z_m(t+1) = \max\{0, Z_m(t) + \mathrm{clip}(d_m(t), 0, 1) - 0.02\}$ is therefore logged without

feeding it into control, together with saturation frequency and physical queue sizes. The V sweep evaluates this implemented heuristic; it is not presented as validation of the $O(1/V)$ proposition.

Proof. Summing Equation (33) over $t = 0, \ldots, T-1$, telescoping the Lyapunov terms, dividing by VT and letting $T \to \infty$ gives the usual drift-plus-penalty bound. Slater slack ensures that the negative drift dominates sufficiently large queues, which implies mean-rate stability. The performance bound follows directly. □

The condition in Proposition 3 is slotwise. More generally, if the action implemented at slot t makes the drift-plus-penalty upper bound at most ϵ_t larger than its slotwise infimum and $\limsup_{T\to\infty} T^{-1}\sum_{t<T}\mathbb{E}[\epsilon_t] \le \bar{\epsilon} < \infty$, the same telescoping argument gives $\overline{U} \ge U^\star - (B+\bar{\epsilon})/V$. An H-step receding-horizon optimiser therefore does not automatically inherit the $O(1/V)$ guarantee. It inherits the approximate bound only when its applied first action is shown to satisfy this per-slot approximation condition.

3.5. Conflict Resolution, Local Execution and Interpretation

When multiple autonomous applications propose discrete policy templates, conflict resolution is modelled over finite action sets P_i. Let $a = (a_1, \ldots, a_N)$ with $a_i \in P_i$ and define the local utilities of Equation (36),

$$u_i(a) = r_i(a_i; \hat{s}_i) - \sum_{j \neq i} \chi_{ij}(a_i, a_j), \tag{36}$$

where χ_{ij} is a symmetric pairwise conflict penalty. The global potential is given by Equation (37),

$$\Phi(a) = \sum_i r_i(a_i; \hat{s}_i) - \sum_{i<j} \chi_{ij}(a_i, a_j). \tag{37}$$

Proposition 4 (Exact potential structure; the standard check of Monderer and Shapley [11])**.** *For a fixed observed state and fixed symmetric pairwise penalties, the finite conflict-resolution game is an exact potential game.*

Proof. For any unilateral deviation of agent i from a_i to a_i', Equation (38) holds:

$$\begin{aligned} \Phi(a_i', a_{-i}) - \Phi(a_i, a_{-i}) &= r_i(a_i'; \hat{s}_i) - r_i(a_i; \hat{s}_i) \\ &\quad - \sum_{j \neq i} \chi_{ij}(a_i', a_j) + \sum_{j \neq i} \chi_{ij}(a_i, a_j) \\ &= u_i(a_i', a_{-i}) - u_i(a_i, a_{-i}), \end{aligned} \tag{38}$$

which is exactly the defining condition of an exact potential game. □

Corollary 1. *For a fixed state and fixed symmetric penalties, strict asynchronous better-response updates over finite P_i terminate in a pure Nash equilibrium [11].*

Proposition 4 establishes the exact-potential structure and Corollary 1 gives the resulting finite better-response convergence property used by the conflict mediator.

A control-theoretic local executor can also be used. When a local domain admits a linearised dynamic model, the fast-timescale executor may instead solve

$$\min_{\{u_i(\tau)\}} \quad \sum_{\tau=t}^{t+H-1} \Big(\big\| C_i \varsigma_i(\tau) - r^t_{i,\mathrm{ref}}(\tau) \big\|^2_{Q_i} + \|u_i(\tau)\|^2_{R_i} + \kappa_i h_i(\tau) \Big) \tag{39}$$

$$\text{s.t.} \quad \varsigma_i(\tau+1) = A_i^{\mathrm{dyn}} \varsigma_i(\tau) + B_i^{\mathrm{dyn}} u_i(\tau) + w_i(\tau), \tag{40}$$

$$\varsigma_i(\tau) \in Z_i, \;\; u_i(\tau) \in U_i, \tag{41}$$

$$\varsigma_i(t+H) \in Z_i^f. \tag{42}$$

The optional local model predictive control (MPC) problem consists of the objective in Equation (39), the dynamics in Equation (40), the state and input constraints in Equation (41) and the terminal constraint in Equation (42).

Here, ς_i is the local state, A_i^{dyn} and B_i^{dyn} are state and input matrices, C_i maps state to controlled outputs and $r^t_{i,\mathrm{ref}}$ is the delegated reference. The notation $\|z\|^2_Q = z^\mathsf{T} Q z$ uses positive semidefinite Q_i and positive definite R_i; $h_i(\tau)$ counts handover-related events for agent i and $\kappa_i \geq 0$ weights their cost. State, input and terminal sets are Z_i, U_i and Z_i^f and w_i is a disturbance. For the nominal model with $w_i = 0$, a control-invariant terminal set, an admissible terminal controller and compatible terminal conditions permit the usual shifted-sequence recursive-feasibility argument [55]. With disturbances, robust constraint tightening or an equivalent invariance construction must also be supplied. This provides a control-theoretic local-execution option whenever a sufficiently accurate plant model is available. The quadratic tracking objective in Equation (39) and the drift-inspired horizon objective of the mobility-aware safe execution (Section 4.4) are alternative formulations. They are not asserted to be equivalent and neither optional horizon solver is used in the reported campaign.

The resulting realisation stack connects typed intent semantics to executable control. Sound refinement maps intent predicates to resource requirements; the slow-timescale optimisation coordinates domain budgets; virtual queues and drift-plus-penalty control address long-run violations; the potential game mediates discrete policy conflicts; and the optional MPC layer supports model-based local execution where suitable plant models are available. This structure directly informs the algorithms presented next. The associated bounded advisory component is described in Section 4.5.

4. Proposed Algorithms and Complexity Analysis

In this section, the executable algorithmic realisation of the formal model is presented. The end-to-end intent flow is described first, from typed ingress through refinement, feasibility certification, delegation, coordination and safety-filtered enactment, together with the messages exchanged in one control epoch. Three algorithms are then specified, namely hierarchical intent refinement and feasibility propagation, trust-aware shared-space primal-dual coordination and mobility-aware safe execution, after which the bounded advisory modulation contributed by the learning core is defined and the computational scaling of the three algorithms is analysed. Each agent runs the perceive, deliberate, commit and execute cycle of Definition 2 and the stage table of that definition locates the cycle in the three algorithms.

Four principles are followed by the algorithms. First, semantics are placed before control, so that every admitted intent must be normalised and typed. Second, control is distributed but governed, so that local autonomy is allowed only inside explicit budgets and trust envelopes. Third, safety is placed before performance, so that an action selected for utility or latency performance is enacted only after local hard constraints, authorisation scope and safety invariants have been checked, and a predefined safe fallback is used otherwise. Fourth, conflicts are treated with awareness, so that application conflicts are detected and mediated explicitly.

Operationally, the three algorithms form one pipeline rather than three independent controllers. Hierarchical intent refinement and feasibility propagation (HIRF) produces the typed child intents, delegated budgets and coupling constraints consumed by trust-aware shared-space primal-dual coordination (TADP). TADP then computes the cross-domain operating point in the common coordination space and returns domain-specific allocations through the local maps. Mobility-aware safe execution (MASE) consumes those allocations together with current telemetry, virtual queues, mobility state and local invariant checks before the first executable action is applied. This ordering prevents a fast executor from silently repairing an infeasible high-level request, because infeasibility is exposed during refinement, while also preventing a mathematically feasible distributed allocation from bypassing local safety checks. The implementation logs the outputs of each stage so that the path from the submitted intent to the enacted action can be reconstructed.

The methodological sequence is staged so that each decision can be checked against the contract produced by the preceding stage. At the semantic stage, the submitted request is converted into typed predicates and explicit KPI targets, which prevents ambiguity from entering the optimisation variables. At the feasibility stage, the complete set of child intents is checked jointly rather than independently, so shared radio, compute, transport and policy couplings are considered before delegation. At the coordination stage, heterogeneous local variables are projected through the maps G_i into the common coordination space, which allows agents with different internal models to exchange only the quantities required for global agreement. At the execution stage, virtual queues represent persistent service pressure, digital-twin predictions provide short-horizon context and local invariants define a non-negotiable admissible action set. Conflict mediation is then invoked only when multiple admissible policy proposals remain mutually incompatible. This separation of concerns is central to the methodology because semantic correctness, joint feasibility, distributed optimality and execution safety are tested at distinct boundaries instead of being collapsed into one opaque policy model.

The resulting implementation can therefore be traced end to end. A submitted intent is first mapped to a canonical object, each refinement edge records the parent-to-child relationship and its delegated budget, each Phase-I solve records whether a zero-slack operating point exists, each PDHG run records its terminal residuals and each MASE decision records whether the candidate action was accepted, projected or replaced by a fallback. The conflict mediator records the pre-mediation proposals and selected admissible profile, while the trust path records the evidence that determined peer admission. These traces are used later in the evaluation to distinguish mechanism availability from actual engagement. The methodology is consequently reproducible not only at the level of final KPI values but also at the level of intermediate decisions that produced those values.

4.1. Hierarchical Intent Flow

The architectural intent-processing flow is summarised in Figure 2. A typed or natural-language intent is submitted, normalised, validated and refined into predicates, budgets and couplings. Resource feasibility is checked before budgets are delegated. Trust admission, coordination and safety filtering precede enactment, after which observed outcomes are returned for assurance. Infeasible requests are returned for policy-governed negotiation. Modelled payloads are listed in Table 7; sizes assume eight bytes per floating-point value. Here, N is the number of admitted resource agents, p is the number of pooled resources, K is the number of coordination iterations and d_i is the local decision dimension. The prototype realises these stages through local calls and simulated admission indicators. The prototype exchanges local objects; the listed volumes do not measure a network protocol.

The flow is staged so that semantic processing is completed before any resource commitment is attempted. At ingress, a submitted intent is first normalised into a canonical typed object so that a common representation is supplied downstream. Schema validation constrains the representation but does not prove that the intended meaning of every free-form request has been recovered. Validation then checks syntax, authorisation, typing and context selectors, so that malformed or unauthorised requests are rejected with an explanation instead of being silently propagated into the optimisation path. Only after these checks are passed is the intent refined into predicates, budgets and couplings, which ensures that the layers responsible for allocation receive an authorised object whose syntax, typing, authorisation and context have been validated. This ordering is what allows correctness to be argued layer by layer rather than re-established globally at every step.

Table 7. Modelled messages and conceptual payloads for one control epoch.

Message	From → to	Payload	Size	Frequency
Typed intent	tenant → business agent	intent object (service type, KPI targets, priority, security level, scope)	≈0.20 kB	on arrival
Child intents and budgets	refinement → domain agents	per child: thresholds, resource floors, coupling row, priority	≈0.10 kB per child	per epoch
Peer proposal	domain agent → coordinator	authentication tag, freshness, scope, consistency and performance evidence, optional advisory factors and requirement request	≈0.10 kB	per epoch per agent
Admission and authorisation	coordinator → domain agent	admission bit, authorisation token, trust score	<0.10 kB	per epoch per agent
Coordination upload	agent → aggregator	$G_i \bar{x}_i \in \mathbb{R}^p$	$8p$ bytes	K per epoch per agent
Coordination download	aggregator → agent	$\lambda \in \mathbb{R}^p$	$8p$ bytes	K per epoch per agent
Certificate and allocation	coordinator → executors	$\rho^\star$, $\zeta^\star$, $x_i \in \mathbb{R}^{d_i}$, templates	$8(1 + p + \sum_i d_i)$ bytes plus template encoding	per epoch
Enactment and verdict	executor → domain agent	applied action, safety-filter verdict, fallback flag	<0.10 kB	per epoch
Telemetry	analytics → registry, twin	per-slice KPIs, energy, handovers, churn	≈0.30 kB	per epoch
Assurance and audit	registry → tenant	fulfilment score, violations, certificate, admissions, verdicts	≈0.50 kB	per epoch

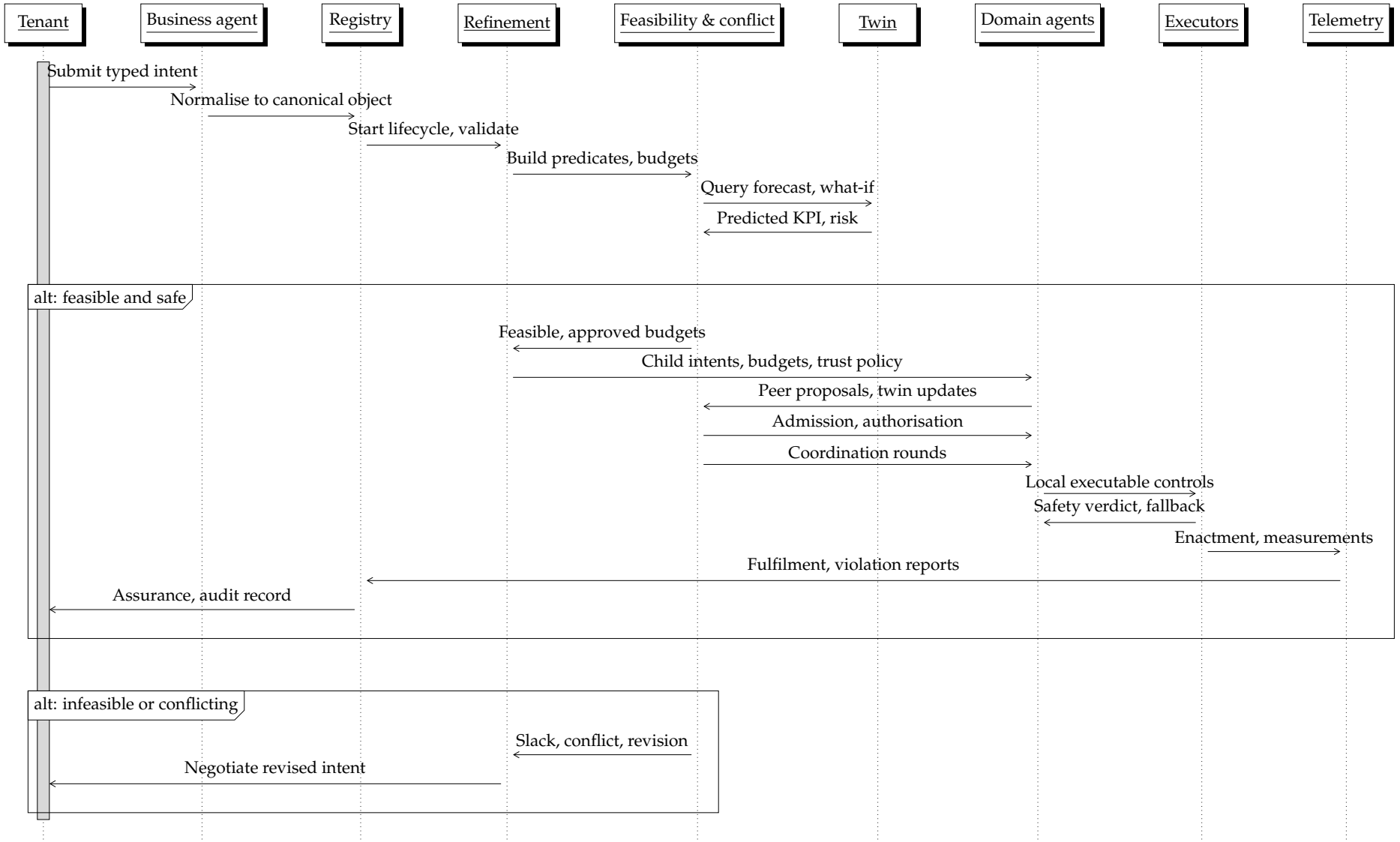


Figure 2. Architectural intent cycle from arrival to verification, including the infeasible negotiation branch.

The central decision point of the flow is the feasibility-and-conflict engine. Before any child intent is delegated, the engine queries the network digital twin for a forecast and a what-if evaluation, so that predicted KPI behaviour and predicted risk are taken into account rather than only the instantaneous state. Two outcomes are then possible. When the request is feasible and safe, the feasibility-and-conflict engine passes the approved resource-budget allocation internally to the refinement engine, which attaches the delegated budgets and trust policy to the corresponding child intents before passing them to the domain agents. The domain agents then produce local executable controls and the resulting enactment and measurements are reported back for assurance and explanation. When the request is infeasible or conflicting, slack information, conflict descriptors and revision options are returned instead and a negotiation exchange with the tenant or operator is opened. This explicit branch is what prevents the system both from over-promising on intents that cannot be satisfied and from failing silently and it is the runtime counterpart of the Phase-I minimum-slack feasibility check and the subsequent zero-slack operating coordination developed in the system model.

Modelled payload sizes for this cycle are summarised in Table 7.

4.2. Hierarchical Intent Refinement

Algorithm 1 is used to normalise and validate the raw intent, generate business-level predicates, refine them into service and then resource intents with coupling constraints and solve the Phase-I minimum-slack feasibility program. If the minimum total slack is zero, the refinement process returns the resulting child intents with attached delegated budgets, KPIs, fallback policies and trust requirements; otherwise the non-zero slack support is used to construct an explainable set of negotiation options.

Algorithm 1 Hierarchical Intent Refinement and Feasibility Propagation (HIRF)

Require: Raw intent input I^{raw}, capability registry $\mathcal{H}$, state estimate $\hat{s}$, trust policy Θ
Ensure: Feasible child intents or negotiated revision
$\iota \leftarrow \mathrm{Normalize}(I^{\mathrm{raw}})$
Validate syntax, authorisation, typing and context selectors
if validation fails **then**
    return Reject with explanation
end if
Generate business-level predicates, contexts and weights
$\{\iota_j^S\} \leftarrow R_{B\to S}(\iota, \hat{s}, \mathcal{H})$
for all service intents ι_j^S **do**
    $\{\iota_{jk}^R\} \leftarrow R_{S\to R}(\iota_j^S, \hat{s}, \mathcal{H})$
    Build coupling constraints and resource budgets
end for
Solve Phase I, Equations (21)–(23)
if $\rho^\star = 0$ **then**
    Solve the zero-slack operating allocation, Equations (24)–(26)
    Attach budgets, KPIs, fallback policies and trust requirements
    return Feasible child intents
else
    Construct negotiation options from the support of $\zeta^\star$
    return Revise or renegotiate
end if

Refinement is treated as the operation that turns a declarative business outcome into a set of executable resource commitments without losing the meaning of the original request. Each refinement step is required to be sound in the sense of Proposition 1, so that fulfilment of the generated children, together with the couplings that they introduce,

implies fulfilment of the parent. This requirement is what allows feasibility to be reasoned about hierarchically rather than re-verified globally at every layer, and it is the reason refinement is presented as a first-class operation rather than as a translation detail. The capability registry is consulted throughout, so that only children which correspond to available domain capabilities are generated, which keeps every produced resource intent implementable rather than aspirational.

The Phase-I minimum-slack program is the point at which joint infeasibility is converted into actionable information. When $\rho^\star = 0$, a zero-slack joint realisation exists for the represented convex surrogate and the child intents may be released with their attached budgets, KPIs, fallback policies and trust requirements after the operating allocation is solved. When $\rho^\star > 0$, the represented child set is jointly infeasible and the support of $\zeta^\star$ identifies the coupling constraints that require the smallest total ℓ_1 relaxation. Negotiation options are constructed from that support rather than from a generic failure signal. A relaxation with minimum total coupling slack is therefore offered to the tenant or operator. Minimising the total slack does not establish that the number of relaxed constraints is minimal. When no human revision is available inside a control epoch, an automated negotiation policy distributes the ℓ_1-minimal excess of each oversubscribed pool over the contributing children; the supplied implementation supports proportional, inverse-priority and headroom-first distribution (children that are comfortably fulfilled are relaxed first) and uses the headroom-first policy by default, which is the choice evaluated in Section 6.4. The dominant cost of the algorithm is the feasibility solve itself and its scaling, together with that of the routines invoked during refinement, is reported in Section 4.6.

4.3. Trust-Aware Distributed Coordination

Algorithm 2 realises the shared-space primal-dual updates in Equations (28)–(30) with authenticated message exchange. Stale, unauthenticated or out-of-scope messages are discarded before their p-dimensional resource contributions can affect coordination.

Algorithm 2 Trust-Aware Shared-Space Primal-Dual Coordination (TADP)

Require: Jointly feasible resource intents, f_i, X_i^t, G_i, budget b, steps τ, σ
Ensure: Candidate allocations $\{\hat{x}_i\}$ with convergence and resource-feasibility status

Initialise $x_i^0 \in X_i^t$, $\lambda^0 \succeq 0$ and trust scores $w_{ij}^0 \in [0,1]$
for $k = 0, 1, \ldots, K-1$ **do**
  for all agents i in parallel **do**
    Receive authenticated, integrity-protected and freshness-checked coordination messages
    Reject expired, replayed, unauthenticated or out-of-scope messages
    Apply the trust-admission rule in Section 5.3
    $x_i^{k+1} \leftarrow \mathrm{prox}_{\tau(f_i + I_{X_i^t})}(x_i^k - \tau G_i^\mathsf{T} \lambda^k)$
    $\bar{x}_i^{k+1} \leftarrow 2x_i^{k+1} - x_i^k$
    Export $y_i^{k+1} \leftarrow G_i \bar{x}_i^{k+1} \in \mathbb{R}^p$
  end for
  Recover the admitted aggregate $y^{k+1} = \sum_i y_i^{k+1}$ in the common coordination space
  $\lambda^{k+1} \leftarrow [\lambda^k + \sigma(y^{k+1} - b)]_+$
  Log hard-gating events; retain the epoch-level soft trust scores
  if declared primal and dual residuals are below tolerance **then**
    break
  end if
end for
$\hat{x}_i \leftarrow x_i^{k+1}$; record whether the stopping criterion was met
Check the candidate resource constraints and record feasibility status
return $\{\hat{x}_i\}$ and the two status indicators

Trust is treated primarily as an admission and weighting mechanism around the convergent base solver, not as a reason to claim convergence from graph connectivity alone. During one coordination epoch, the admitted-agent set and local objective coefficients are held fixed. Proposition 2 applies when the aggregate in Equation (30) is recovered exactly at each solver iteration. The supplied implementation computes this sum through local calls. A distributed averaging implementation requires a separate analysis of aggregate recovery and its errors. Connectivity and an aggregation error that only vanishes asymptotically do not, by themselves, establish the stated PDHG convergence result. Persistent admission changes or inexact aggregation are outside the guarantee stated here.

Trust rejection can violate Assumption 4 when admitted peers lose connectivity. The global convergence proposition then ceases to apply. A connected component can solve a local problem only if an explicit component budget is assigned and all component budgets sum to no more than the original global budget; independently giving each component the complete budget would violate the coupling constraint. No component-partition solver is implemented in this package. Instead, connectivity is monitored and a disconnected epoch reuses a copy of the last resource-certified action, or a balanced initial action if none exists. Current hard share, power and security limits are checked again. A certificate from a previous epoch is not treated as proof of current intent feasibility, so the fallback is explicitly labelled uncertified for the current surrogate and no stale solver cost is copied into that epoch. Recovery re-enables ordinary coordination. A regression trace exercises initial disconnection, disconnection after successful operation and recovery. The campaign graph retains a trusted coordinator hub adjacent to every admitted domain peer, so rejection alone does not disconnect that graph; the deliberate edge-removal test is separate evidence for the fallback branch. In the analytical implementation the trust gate controls advisory peers and poisoned twin inputs rather than removing the resource-slice agents from the optimiser.

This formulation also resolves the dimensional-compatibility issue. An agent never subtracts a heterogeneous decision vector $x_j \in \mathbb{R}^{d_j}$ from $x_i \in \mathbb{R}^{d_i}$. It exchanges or aggregates $G_i x_i \in \mathbb{R}^p$, together with residual information in the same common resource space. Soft trust scores are updated once per control epoch in the supplied implementation, while hard authentication, freshness and authorisation failures cause immediate exclusion as described in Section 5.3. The iteration stops when the implemented residuals fall below the declared tolerance or the iteration limit is reached. The logged primal residual is $\|[\mathrm{K}x - b]_+\|_2$ and the logged dual residual is $\|\lambda^{k+1} - \lambda^k\|_2/\sigma$. These are a feasibility residual and a dual-step diagnostic; they do not constitute a complete stationarity certificate or a primal-dual objective gap.

4.4. Mobility-Aware Safe Execution

Fast-timescale actuation is specified in Algorithm 3. In the evaluated implementation, capped deficits are updated once per epoch and used in coordination, after which mediated templates are assembled with the resource budgets and passed through a hard safety filter. An optional MPC branch may instead be used to predict and optimise H-step trajectories. No such horizon optimisation is executed in the reported analytical campaign.

Execution operates at the fastest timescale of the architecture, where decisions must be taken more quickly than slow-timescale coordination can be repeated. A dedicated execution layer is therefore required, so that local actuation can track mobility and load variation without waiting for a fresh global allocation. In the evaluated mode, capped deficit states are maintained for the protected predicates using Equation (35). Persistent deficits can therefore influence subsequent coordination, while overfulfilment can reduce these states. The ideal uncapped queues of Equation (31) are reserved for the theoretical execution option. The mobility state of each user is observed explicitly, so that handover-

related actuation is informed by position, velocity and contextual mode rather than by instantaneous signal strength alone.

Algorithm 3 Mobility-Aware Safe Execution (MASE)

Require: Delegated intent ι_i, budgets, queue state, mobility state $\xi_u(t)$, safety set $\mathcal{U}_i^{\text{safe}}$, execution mode
Ensure: Safe local actuation $u_i(t)$
for each execution slot t **do**
  Observe $\hat{s}_{i,t}$, $\xi_u(t)$ and current budgets
  if evaluated template mode **then**
    Use capped deficits updated once this epoch by Equation (35)
    Assemble $u_i^\star(t)$ from coordinated budgets and mediated templates
  else
    Update ideal queues by Equation (31); predict H-step trajectories
    Select the first action of the optional receding-horizon solution
    $$\arg\min_{u_{t:t+H-1}} \sum_{\tau=t}^{t+H-1} \Big(\sum_m Q_m(t) v_m(\tau) - V U_\tau + \kappa_i h_i(\tau) \Big)$$
    subject to local safety and budget constraints
  end if
  Apply hysteresis and anti-ping-pong guard if association-related
  if $u_i^\star(t) \notin \mathcal{U}_i^{\text{safe}}$ **then**
    $u_i(t) \leftarrow \text{FallbackSafeAction}(\hat{s}_{i,t})$
  else
    $u_i(t) \leftarrow u_i^\star(t)$
  end if
  Enact $u_i(t)$
  Log decision, explanation and observed outcome
end for

In the optional MPC branch, utility is balanced against accumulated backlog in a receding-horizon problem and is motivated by the same drift-plus-penalty structure. It does not automatically inherit the $O(1/V)$ result in Proposition 3, because that proposition assumes slotwise minimisation of the drift bound. The bound applies to the receding-horizon implementation only if the first action actually applied at every slot satisfies the exact or bounded-error per-slot condition stated after Proposition 3; otherwise the H-step controller is treated as an MPC-inspired execution heuristic with its own safety and recursive-feasibility conditions. Hysteresis and an anti-ping-pong guard are applied to association-related actions, so that rapid oscillation between cells, which would otherwise inflate signalling and degrade the user experience, is suppressed. The final safeguard is the local safety override. Whenever the optimised action would leave the safe set, it is replaced by a fallback action that is known to respect the hard invariants, so that only an action admitted by the local safety mechanism is enacted. Every decision, together with its explanation and observed outcome, is logged, thereby supporting the auditability of the execution layer in accordance with the security requirements stated later in the paper.

4.5. Bounded Advisory Modulation by the Learning Core

The bounded advisory component is called the fuzzy learning core in the ablations of Section 6. Its learned belief predictors and fixed fuzzy rule base are distinguished below; the digital twin also contains learned response estimates. It sits in the deliberation stage of the domain coordinator and does not make decisions by itself: it emits, per slice, an advisory factor $\alpha_j \in [0.70, 1.40]$ that scales the priority weight of that slice in the operating objective of the current epoch. The factor is produced in two stages. Candidate regressors are trained by a belief predictor on the agent's own rolling window of six epochs and

ranked by cross-validated skill. The best regressor is selected to forecast each slice's backlog pressure and latency ratio one epoch ahead. The forecasts are recorded as predictive beliefs. A fixed Mamdani fuzzy plan [56] then maps the two predicted inputs, backlog pressure relative to the latency budget and realised-over-target latency ratio, each described by three linguistic terms (trapezoidal at the extremes, triangular in the middle), through nine expert rules with four consequents (release, hold, raise, surge), min conjunction, max aggregation and centroid defuzzification, to a share adjustment. An increase is recommended when a slice is backlogged or breaches its latency target. A reduction is recommended only when both inputs are low; otherwise, the current setting is retained. No parameters are trained in the rule base, so the plan itself has no learning lag. The adjustment is mapped to the advisory factor through the acceptance envelope $\alpha_j = \mathrm{clip}(1 + 0.60\,(f_j - 1),\, 0.70,\, 1.40)$, where f_j is the defuzzified adjustment. Two relevant properties are preserved by the bounded advisory. The acceptance envelope is enforced at the interface, so the modulated objective remains strictly convex with bounded coefficients. The coefficients are held fixed during each solve and the other assumptions of Proposition 2 remain necessary. The typed floors of the local boxes X_i^t are not modified by the advisory, so a certified child set cannot be rendered infeasible through this weight adjustment. Peer coordination messages may carry advisory proposals through the same interface; for this reason the trust admission rule and the acceptance envelope are evaluated in the adversarial-coordination scenario of Section 6. The ablation "no fuzzy learning core" fixes every α_j to one.

4.6. Complexity Analysis

Indicative computational scaling for the proposed algorithms and supporting routines is summarised in Table 8. The symbols are defined explicitly because solver complexity is a function of the total decision and constraint dimensions, not only of one local vector. Let n_F and m_F be the total variable and constraint dimensions of the Phase-I feasibility problem, let p be the common coupling-space dimension, let n_i^{MPC} and m_i^{MPC} be the horizon-stacked local MPC variable and constraint dimensions and let $\bar{P} = \max_i |P_i|$. The iteration counts of an interior-point method and TADP are denoted by N_{IP} and K, respectively.

Table 8. Solver-dependent computational scaling of the proposed algorithms.

Algorithm	Dominant Work	Scaling Under the Stated Assumption
HIRF/Phase I	Parsing, template expansion, capability lookup and joint minimum-slack solve.	Parsing is near-linear in the intent representation. Under a dense unstructured interior-point factorisation, a conservative dense-factorisation estimate is $O\big(N_{\mathrm{IP}}(n_F + m_F)^3\big)$; sparse structure can reduce this substantially.
TADP	Local proximal/gradient work, multiplication by G_i and aggregation in the p-dimensional coupling space.	$O\Big(K\sum_i(c_i^{\mathrm{prox}} + \mathrm{nnz}(G_i)) + Kp\Big)$ arithmetic plus authenticated aggregation/communication, where c_i^{prox} is the cost of the local proximal step.
MASE, optional MPC branch	Horizon-stacked local optimisation and hard safety filter.	For a dense unstructured quadratic-program (QP) factorisation, $O\big(N_{\mathrm{IP}}(n_i^{\mathrm{MPC}} + m_i^{\mathrm{MPC}})^3\big)$ per solve. Sparse or banded MPC methods can scale much more favourably with H when local state and control dimensions are fixed.
Conflict game	Evaluation of conflicting template pairs on the conflict graph $\mathcal{E}_c$.	$O\Big(\sum_{(i,j)\in\mathcal{E}_c} \lvert P_i\rvert\lvert P_j\rvert\Big)$; a dense upper bound is $O(N^2\bar{P}^2)$.
Trust/security	Authentication, evidence scoring, admission and audit updates.	Linear in the number of received coordination messages for the scoring logic, plus the implementation-dependent cost of the selected cryptographic primitives.

Abbreviations: hierarchical intent refinement and feasibility propagation (HIRF); trust-aware shared-space primal-dual coordination (TADP); mobility-aware safe execution (MASE); model predictive control (MPC).

These expressions are scaling relationships under explicit solver assumptions rather than universal worst-case bounds. In particular, $O(Hd_i^3)$ is not used because a horizon-stacked MPC problem grows in both variables and constraints with H. Likewise, pairwise conflict testing depends on combinations of the template sets of both agents, so a single factor $\bar{P}$ is insufficient in the dense case. Deployment-level runtime measurements remain necessary because sparsity, warm starts and solver choice can dominate the practical cost.

5. Security, Privacy and Trust

The security, privacy and trust model that constrains every agentic control action is established in this section. The threat model instantiated in the evaluation is defined first, with the capabilities of the adversary and the components that are assumed trusted. The security and trust requirements that every agentic action must satisfy are then established and the trust-scoring and admission model is specified as implemented, with its evidence indicators, its update rule and its hard admission gate. Finally, the privacy-preserving deployment options by which raw telemetry is kept inside the domain that produced it are described.

5.1. Security Threat Model

Agentic intent-based networking introduces a broader attack surface than conventional network-management systems. The following threat classes are considered: intent injection through malformed or adversarial user or operator inputs; telemetry spoofing and stale digital-twin state; model poisoning or corrupted federated updates; conflict exploitation involving xApps and rApps to induce performance degradation; agent compromise, including tool, prompt or memory compromise; and the classical 5GS threats that the underlying mobile system already faces.

The breadth of this list reflects the fact that an agentic intent-based system exposes attack surfaces that a classical management system does not. Three groups can be distinguished for clarity. The first group targets the semantic layer, where intent injection through malformed or adversarial inputs attempts to make the system pursue a goal that the operator never sanctioned. The second group targets the information on which decisions are based, where telemetry spoofing and stale digital-twin state are used to corrupt the perceived network condition and where model poisoning corrupts the learned components through tampered federated updates. The third group targets the agents themselves, where xApp or rApp conflict exploitation, tool, prompt or memory compromise and the underlying classical 5GS threats are used to subvert execution directly.

This threat model is not catalogued for its own sake but is used to derive the requirements and mechanisms that follow. Intent injection is mitigated through typed, validated and authorised ingress, so that only well-formed and sanctioned objects enter the optimisation path. Telemetry and digital-twin corruption are mitigated through authenticated, integrity-protected and freshness-checked messaging, while trust scoring controls the influence of authenticated but potentially unreliable peers. Agent compromise is addressed through sandboxed evaluation of new control applications, access-controlled audit logging and safe local fallbacks, which support the detection, isolation and containment of a compromised component. Classical 5GS threats remain addressed by the baseline telecom security architecture, which is complemented rather than replaced by the agentic-specific safeguards.

The threat model instantiated in the evaluation is the following. The adversary controls a fraction of the five advisory peers (20% to 60% in the sensitivity study, 40% in the campaign) for persistent episodes and through them can (i) report spoofed telemetry, (ii) push poisoned updates into the shared digital twin, (iii) attach unrestrained advisory

proposals and resource-grab requests to coordination messages and (iv) collude, in the sense that all compromised agents pursue the same objective (starve the highest-priority slice, inflate another). The adversary cannot forge the identity of an honest agent, cannot compromise the coordinator hub or the intent registry and cannot alter the typed floors or the Phase-I certificate, which are not exposed on the coordination channel; key compromise is out of scope. Under this model the trust gate and the bounded advisory envelope restrict access to the operating objective; for this reason its admission decisions, its false rejections and its reaction time are measured directly in Section 6.5.

5.2. Security and Trust Requirements

The following requirements are imposed by the architecture.

S1: Typed and authorised intent ingress.
S2: Authenticated, integrity-protected and freshness-checked inter-agent messages.
S3: Trust-weighted acceptance rather than blind acceptance of authenticated messages.
S4: Sandboxed evaluation and profiling of new control applications where feasible.
S5: Access-controlled, proportionate and tamper-evident audit logging of refinement, negotiation, coordination, conflict and actuation events.
S6: Safe local fallbacks in every execution domain.
S7: Short-lived, least-privilege capability tokens that restrict each agent to declared domains, resources, tools and action classes.
S8: Runtime validation of learned or LLM-derived candidate actions before they reach the actuation interface.

Each requirement is a direct response to the threat model. The intent-injection surface is reduced through typed and authorised ingress, while semantically harmful requests that satisfy the schema still require policy checks. Authenticated, integrity-protected and freshness-checked inter-agent messaging enables replayed, modified and spoofed coordination traffic to be rejected before it affects an update. Trust-weighted acceptance is required in addition to authentication, because a message can be cryptographically valid and still originate from a compromised or malfunctioning agent, so authenticity and trustworthiness are deliberately separated as distinct properties. Sandboxed evaluation and profiling allow the behaviour of a new control application to be assessed in isolation before it is granted influence over live resources. Access-controlled and tamper-evident audit logging supports the reconstruction of relevant refinement, negotiation, coordination, conflict and actuation events for accountability and offline analysis. Safe local fallbacks provide each execution domain with a predefined admissible action when higher layers are degraded or unavailable.

These requirements are intended to be enforced jointly rather than selectively, so that no single mechanism is treated as sufficient on its own. The design follows a defence-in-depth principle in which the semantic layer, the coordination layer and the execution layer each retain an independent safeguard. A failure or compromise at one layer can thus be constrained by safeguards at another, while the control algorithms operate under the explicit assumption that admitted information satisfies the stated authentication and trust conditions. Security is consequently treated as an architectural property of the whole framework rather than as a feature attached to a single component.

Quantum-safe protection is separated by function. Post-quantum cryptography (PQC) and quantum key distribution (QKD) are used for different purposes. Quantum-resistant key establishment is specified by the National Institute of Standards and Technology (NIST) Module-Lattice-Based Key-Encapsulation Mechanism standard [57]. Digital signatures are specified separately by the Module-Lattice-Based Digital Signature standard [58] and the Stateless Hash-Based Digital Signature standard [59]. Shared symmetric key material can

be generated and delivered through QKD infrastructure, but entity authentication, authorisation and application-data encryption must be provided separately [60,61]. Application confidentiality and integrity are provided by symmetric authenticated encryption after key establishment. These are deployment options; the analytical experiments use simulated security indicators.

The threat-to-mechanism mapping is therefore explicit. Prompt injection and unsafe LLM output are addressed by typed ingress, schema validation, output validation, the least privilege, sandboxing and the local runtime verifier, consistent with current Open Worldwide Application Security Project (OWASP) GenAI guidance [32] and the NIST Generative AI risk-management profile [31]; a watermark or provenance tag may support attribution or tamper evidence but is not treated as a prompt-injection defence. Federated-update interception is addressed by an authenticated channel whose traffic keys are established using deployment-appropriate PQC, QKD-derived symmetric material or a hybrid construction. Model poisoning is not solved by channel encryption and remains subject to provenance, aggregation checks and trust controls. Fake or unauthorised agents are addressed by identity authentication and capability-based authorisation rather than by QKD itself.

Runtime privilege is represented by a capability envelope $\Gamma_i(t) = (\mathrm{id}_i, \mathcal{D}_i, \mathcal{R}_i, \mathcal{T}_i, \mathcal{A}_i, t_{\mathrm{exp}}, \mathrm{nonce})$, which identifies the agent, authorised domains $\mathcal{D}_i$, resources $\mathcal{R}_i$, tools $\mathcal{T}_i$, action classes $\mathcal{A}_i$ and credential expiry. An action outside this envelope is rejected before actuation. The envelope can be carried in a short-lived signed or message authentication code (MAC) protected capability token. PQC signatures or an existing public key infrastructure (PKI) can authenticate the identity and token, while QKD-derived material may protect a symmetric channel or token in transit when suitable infrastructure exists. The envelope defines an authorisation scope for permitted actions and resources.

No universal microsecond lifetime is assigned to QKD material. Key lifetime, refresh and consumption are implementation and policy choices and a one-time-pad construction is distinct from using QKD-derived material as a Transport Layer Security (TLS) or authenticated encryption with associated data (AEAD) key. Public certificates are also distinct from private authentication keys: QKD does not remove the need for persistent identity credentials or an authentication infrastructure. Short-lived session keys and capability tokens can reduce exposure, but their validity interval must be selected from the deployment threat model and protocol timing requirements.

In an optional deployment pattern, a fresh QKD-derived pre-shared key could be requested from an interoperable key-management service for each federated round. The key could then be supplied to a compatible TLS 1.3 pre-shared-key handshake. Record-protection keys would be derived from traffic secrets through the TLS key schedule; exporters provide separate application-specific keying material [62]. Key expiry and round cancellation would follow the configured consumption policy. This pattern is specified for deployment and is not exercised by the proof of concept.

5.3. Trust Model

Let $w_{ij}(t) \in [0, 1]$ denote the soft behavioural trust score that agent i assigns to agent j. All soft evidence terms are normalised to $[0, 1]$ and combined as

$$e_{ij}(t) = a_{\mathrm{auth}}\phi_{\mathrm{auth}}(t) + a_{\mathrm{cons}}\phi_{\mathrm{cons}}(t) + a_{\mathrm{perf}}\phi_{\mathrm{perf}}(t) + a_{\mathrm{safe}}\big(1 - \phi_{\mathrm{viol}}(t)\big), \tag{43}$$

$$w_{ij}(t+1) = \alpha_w w_{ij}(t) + (1 - \alpha_w)e_{ij}(t), \tag{44}$$

The bounded evidence score in Equation (43) is filtered through the exponentially weighted moving average (EWMA) reputation update in Equation (44).

Every coefficient is non-negative and $a_{\text{auth}} + a_{\text{cons}} + a_{\text{perf}} + a_{\text{safe}} = 1$. Here, ϕ_{auth} represents soft authentication and freshness quality after hard protocol checks, ϕ_{cons} measures consistency with independently observed telemetry or twin predictions, ϕ_{perf} measures whether accepted proposals delivered their declared outcome and ϕ_{viol} is a normalised responsibility or safety-violation score. The same definitions, ranges and weights must be fixed before a comparative experiment and used for every controller that consumes this trust model.

Gradual reputation decay is not used for hard security failures. Define the hard admission gate of Equation (45),

$$g_{ij}(t) = \mathbb{1}\{A_{ij}(t) = 1, F_{ij}(t) = 1, S_{ij}(t) = 1, I_{ij}^{\text{safe}}(t) = 1, w_{ij}(t+1) \geq \underline{\theta}_{ij}\}, \tag{45}$$

In the simulator, reputations are updated once per epoch with $\alpha_w = 0.70$, initial value 0.80 and threshold 0.60. The evidence weights are $(0.20, 0.40, 0.15, 0.25)$ for authentication, consistency, performance and safety. Hard checks are represented by simulated Boolean indicators, not an implemented cryptographic protocol. The consistency, performance and violation indicators are synthetic random variables whose distributions depend on the attack scenario. They are not measurements produced by an independently validated attack detector. The gate itself consumes only those indicators; missing soft indicators receive the same neutral value regardless of the ground-truth compromise label. Ground truth is used for attack injection and evaluation counts. Per-peer labels, scores, hard-check status and admission decisions are exported. Reaction time is measured from each peer's compromise onset to its first rejection during that episode, with undetected episodes censored at the end of observation. False rejections of honest peers are counted separately. Because an attacker may mimic honest evidence, no universal one- or two-epoch detection guarantee follows from the EWMA update. The sensitivity results and a controlled honest-evidence mimic test quantify this limitation.

A_{ij} denotes successful endpoint authentication and integrity verification, F_{ij} is freshness and anti-replay validity, S_{ij} indicates that the message and requested action lie inside the authorised capability envelope and I_{ij}^{safe} indicates that the requested action violates no hard safety invariant of the execution domain. A failed signature or MAC, replay, expired credential, scope violation or hard safety-invariant violation sets $g_{ij} = 0$ immediately irrespective of the historical reputation score; the hard safety check is thus part of the admission gate itself and is not deferred to the local safety filter. The exponential moving average is reserved for uncertain behavioural evidence, for which α_w controls the memory of past observations.

The trust score is consumed only after hard admission has succeeded. The implementation updates the reputation before applying the gate: at epoch t the evidence $e_{ij}(t)$ is first folded into $w_{ij}(t+1)$ by Equation (44) and the admission decision of epoch t is then taken by Equation (45) with that updated value, which is why Equation (45) is written with $w_{ij}(t+1)$. In TADP, admitted messages may be used in the common coordination aggregate, while messages with $g_{ij} = 0$ are excluded and logged. This separation prevents a previously high reputation from overriding an objective authentication or authorisation failure and makes the trust model compatible with the least-privilege runtime controls defined above.

5.4. Privacy-Preserving Options

To avoid centralising sensitive raw telemetry, deployment options include local analytics with aggregated exports only; federated learning across RICs or domain agents [53]; secure aggregation and, where needed, differential privacy; reputation-backed accountabil-

ity for cross-domain collaboration [54]; and selective logging with auditable summaries instead of unrestricted telemetry replication.

Privacy is treated as a first-class concern, because the raw telemetry that is generated by a mobile network is intrinsically sensitive. Detailed traces can expose user mobility patterns, the composition of deployed services and commercially sensitive topology and the data-locality argument made earlier in the paper is precisely the reason a centralised collection of such data is undesirable. The listed options provide deployment-dependent mechanisms through which the architecture can avoid creating a single concentration point for sensitive data.

These mechanisms are intended to be composed according to the sensitivity of the data and the trust relationships between domains. Local analytics with aggregated exports keeps raw data inside the domain that produced it and shares only summaries. Federated learning across RICs or domain agents allows shared models to be trained without directly centralising the underlying raw telemetry. Secure aggregation and, where the residual disclosure risk is still unacceptable, differential privacy further limit what can be inferred from the exchanged updates. Reputation-backed accountability supports attribution when cross-domain collaboration involves partially trusted parties, while selective logging and auditable summaries support the security model's accountability requirements without unrestricted telemetry replication. The privacy posture is consequently adjustable, and its strength is one of the open design parameters that are left to the operator.

6. Proof-of-Concept Evaluation

In this section, the evaluation endpoints are defined, the platforms, scenario families and comparators are specified, the archived legacy bridge records are reported with their provenance and the analytical hold-out campaign is presented together with the auxiliary experiments, the mechanism traces, the measured costs and a coordination-layer benchmark of the shared-space mechanism, before the scope and limitations of the evidence are stated. The completed analytical studies have effective configurations, seed manifests and source hashes. The recovered bridge records are reproducible workflow dry runs and are not represented as packet-level measurements. Broader platforms and interfaces in the evaluation blueprint are possible validation routes rather than completed implementations. The scope of the reported evidence is defined by the retained configurations and run records.

6.1. Evaluation Philosophy and Intent-Fulfilment Metrics

A study of the framework must evaluate four properties together: semantic correctness of intent handling, closed-loop performance, robustness under adversarial and degraded conditions and the cost of coordination. The compound endpoint is defined in Equation (46).

Suggested evaluation metrics are organised by class in Table 9, so that each layer of the architecture is associated with the quantities that make its behaviour observable.

For time-based reporting, a violation duration is the contiguous-elapsed time from the first slot in which an intent predicate is violated until the predicate is restored to its admissible region for the declared recovery-hold condition. It is therefore a recovery metric, not a synonym for conflict-detection time. Faster conflict mediation may reduce violation duration when a policy conflict is the cause, but the duration can also be dominated by mobility, capacity shortage, attack mitigation or actuator dynamics.

Table 9. Suggested evaluation metrics.

Metric Class	Examples
Intent layer	Parsing success, typed-schema accuracy, refinement soundness, feasibility rate, negotiation rounds, fulfilment ratio, weighted violation score.
Network layer	Mean and tail latency, throughput, goodput, spectral efficiency, packet loss, reliability, handover-failure rate, ping-pong ratio, compute utilisation, slice-isolation leakage, fairness.
Control layer	Convergence time, primal/dual residuals, oscillation index, control regret, configuration changes per hour, communication overhead.
Security and trust	False accept/reject in trust gating, conflict-detection time, violation duration, degradation under spoofing or poisoning, attack-success rate.
Energy and sustainability	Energy per bit, active-cell ratio, MEC energy, KPI per Watt.

Abbreviations: multi-access edge computing (MEC).

The five rows in Table 9 are cross-cutting evaluation classes, not architectural layers; the architecture remains the four-layer structure of Section 3.1. Reporting all five classes together is what exposes a gain at one layer that has been purchased at another.

For the comparative study, an author-defined intent-fulfilment score is used as the primary aggregate endpoint. Its components are declared explicitly because the controllers have different local objectives. The score is defined by

$$S_{\mathrm{IF}} = \sum_{m \in \mathcal{M}} \omega_m \left(1 - \min\{v_m / v_m^{\max}, 1\}\right) - \alpha_E \tilde{E} - \alpha_H \tilde{H} - \alpha_C \tilde{C} - \alpha_S \tilde{R}_{\mathrm{sec}}, \tag{46}$$

where the first term rewards fulfilment of the weighted KPI predicates that the intent specifies and the remaining normalised terms penalise energy use, mobility overhead, configuration churn and security risk. For every reported comparison, $v_m^{\max} > 0$ is selected before evaluation from the corresponding intent scale or an explicitly declared engineering tolerance, $\hat{v}_m = \min\{v_m / v_m^{\max}, 1\} \in [0, 1]$, $\omega_m \geq 0$ and $\sum_m \omega_m = 1$. The cost terms $\tilde{E}$, $\tilde{H}$, $\tilde{C}$ and $\tilde{R}_{\mathrm{sec}}$ are also scaled to $[0, 1]$. The coefficients α_E, α_H, α_C and α_S are non-negative and are fixed before the controller comparison rather than tuned separately for each method. Under these conventions, $S_{\mathrm{IF}} \in [-\alpha_\Sigma, 1]$, where $\alpha_\Sigma = \alpha_E + \alpha_H + \alpha_C + \alpha_S$. In the supplied simulation package the four penalty coefficients are 0.08, 0.05, 0.05 and 0.08, respectively, so $\alpha_\Sigma = 0.26$ and the same KPI weights, normalisation constants and cost coefficients are used for every compared controller. Two clarifications are recorded for transparency. First, the archived legacy bridge table in Section 6.3 was produced by the archived version of the ns-3 bridge whose endpoint was a priority-weighted fulfilment-only score of the throughput, latency, reliability and security predicates without the four cost penalties; that archived column is thus labelled as such and is not numerically comparable with the S_{IF} values of the multi-scenario study in Section 6.4. Second, the supplied current bridge is configured to compute the S_{IF} defined here; the recovered records do not exercise the updated S_{IF} computation at packet level (the earlier score is retained in the exported logs only as an audit column). The individual KPIs remain necessary for diagnosis. Throughput is the served traffic rate in Mbps and latency is the analytical end-to-end delay in ms. Reliability and loss are model-generated indicators, not complementary packet-count fractions. Their implemented relationship is specified in Equations (47) and (48):

$$R_s^{\mathrm{model}} = \mathrm{clip}_{[0,0.9995]}\left[\left(0.55 + 0.448\,\sigma(z_s/5)e^{-1.2\ell_s^+} + 0.04h_s^{\mathrm{sec}}\right)(1 - 0.6d_s^{\mathrm{drop}})\right], \tag{47}$$

$$L_s^{\mathrm{model}} = d_s^{\mathrm{drop}} + (1 - R_s^{\mathrm{model}})(1 - d_s^{\mathrm{drop}}). \tag{48}$$

Here z_s is the slice SINR in dB, $\sigma(x) = 1/(1+e^{-x})$, $\ell_s^+ = \max\{0, (D_s - D_s^{\max})/D_s^{\max}\}$, h_s^{sec} is the configured hardening level and $d_s^{\text{drop}} \in [0,1]$ is the clipped buffer-drop ratio. In code, that ratio is dropped traffic in Mbit divided by the larger of fresh arrivals and 1 Mbit. The coefficients are author-selected analytical-model parameters. These definitions explain why the reported reliability and loss columns do not sum to one and require independent calibration before being interpreted as packet-level reliability estimates. In the main analytical KPI table, the p95 latency is the 95th percentile of slice-epoch latencies pooled over the five families and twenty seeds. In per-run summaries, the percentile is computed within each run before aggregation. Neither quantity is a packet-delay percentile. Violation fraction is the share of epochs above a target and episode duration is measured in consecutive epochs. Configuration churn is the mean absolute change in pooled resource shares. Twin error is the mean across slices of absolute throughput forecast error divided by $\max(1\,\text{Mbps}, |T|)$, where T is realised throughput, as implemented in the supplied code. The stored forecast is the most recent what-if prediction; its association with the finally applied action must be checked before this diagnostic is treated as independent predictive validation. Peak resident set size (RSS) measures process memory; a kibibyte (KiB) is 1024 bytes and a mebibyte (MiB) is 1024^2 bytes. Confidence intervals (CIs), paired effects and multiplicity corrections are specified in Section 6.4. The score is used to compare compound intent fulfilment under the declared weights. Fairness also depends on the comparator implementations, their tuning budgets and the evaluation protocol.

6.2. Platforms, Datasets, Scenario Families and Open Parameters

The experimental platforms and the data used are described first, then the scenario families and the comparators against which the proposed controller is evaluated and finally the design parameters that are left as configurable choices, together with the values used in the reported campaigns.

6.2.1. Experimental Platforms and Datasets

Synthetic traffic, mobility and attack realisations generated from declared seeds are used in the completed analytical experiments. No existing public dataset is used to produce or confirm the reported results. Raw analytical records, effective configurations, seed manifests, source hashes, dependency versions and analysis scripts are retained in the source package. A separate legacy bridge archive contains its configuration, three per-run summaries and per-flow records; its dry-run provenance is assessed in Section 6.3. The data and source code are available upon request to the authors.

Further validation could be conducted with ns-3 5G-LENA for radio-level simulation [63], ns-O-RAN for simulated O-RAN interfaces [64], Simu5G for protocol-level studies [65] and Colosseum for emulation [66]. Integration with a radio access network intelligent controller could be tested through the documented srsRAN interface [67]. Public traces could be supplied through analytics interfaces in future experiments; no such ingestion is claimed for the present runs.

Candidate public resources include DeepMIMO, through which site-specific millimetre-wave and massive multiple-input multiple-output (MIMO) channels can be generated [68]. Traffic variation could be evaluated with the Madrid Long-Term Evolution (LTE) traces [69]. Infrastructure-wide intent-based data are provided by IW-IB-5GNET [70], while data collection and closed-loop experimentation are supported by OpenRAN Gym [71]. These resources are possible extensions of the evaluation, not datasets used to obtain the reported results.

The supplied bridge targets ns-3 5G-LENA as a route to packet-level validation. The recovered bridge outputs reported below are dry-run workflow records. The remaining platforms are routes towards broader validation.

6.2.2. Scenario Families and Baseline Comparators

The scenario families used here are nominal multi-slice operation, URLLC assurance under mobility with recurring surges, conflict-rich O-RAN control, trust-degraded coordination (spoofed telemetry, stale twin, poisoned updates), multi-domain contention, adversarial coordination and sustained overload; each is designed to place a different mechanism under stress. The first five are the principal scenario families used for the main campaign, whereas the two additional stress families (adversarial coordination and sustained overload) are reported separately.

The comparator families were selected to cover threshold control, intent refinement, learning, conflict mediation, workflow orchestration and trust handling. The authored comparator adaptations are described in Table 10. In the result tables, the row labels Central solver, PACIFISTA-style (tuned), PACIFISTA-style (untuned) and Secure agentic IBN denote the central full-information solver, the tuned and untuned PACIFISTA-style comparators and the MARE-style secure agentic IBN comparator; PACIFISTA-style and MARE-style indicate authored adaptations and not the published implementations. The cited papers motivate the method families; the measured results belong to the supplied implementations. The tabular learning, load-based slicing, rule-based workflow and security heuristics are not reproduced implementations of the cited deep-learning, GNN, LLM or MARE systems. The same exogenous traffic, mobility and attack seeds are supplied within each paired block. Realised telemetry may differ because earlier controller actions change the network state.

Table 10. Comparator families and authored adaptations used in the analytical campaign.

Comparator	Representative Source	Implemented Behaviour in Simulation	Reason for Inclusion
Static policy and threshold control	Operator-rule baseline	Fixed thresholds for admission, handover, scaling and energy-saving decisions.	It represents the non-agentic operational baseline against which autonomy should be justified.
Centralised policy engine with typed or LLM ingress	3GPP/ETSI-style closed-loop baseline and LLM-ingress work [5,41]	Natural-language or typed intents are validated into typed objects, after which a single central policy engine applies balanced templates and demand-proportional shares without distributed coordination. A centralised orchestrator with typed ingress and an LLM-extraction front end that feeds the same policy engine coincide in this implementation and are therefore one comparator.	It separates the value of typed intent handling from the value of coordination.
Central full-information solver	This work (comparator)	A single orchestration entity holds every local model and solves the identical operating surrogate centrally to a numerical tolerance (sequential least-squares programming (SLSQP) [72]), with the same twin estimates and typed floors, then pushes the allocation with the intents' preferred templates; no distributed iteration, peer admission, virtual queues, negotiation or template mediation.	It compares the full proposed stack with a central solver of the same surrogate; the other disabled mechanisms prevent solver-only causal attribution.

Table 10. *Cont.*

Comparator	Representative Source	Implemented Behaviour in Simulation	Reason for Inclusion
Hierarchical IBN refinement	Gomes, Gritli and Xie-style intent decomposition [35–37]	Business intents are decomposed into service and resource intents, but local agents do not use trust-aware distributed optimisation.	It tests whether the proposed mathematical coordination layer adds value beyond hierarchical decomposition.
Tabular MARL template control	Sana et al.-style distributed handover and association [43,44]	Independent tabular Q-learning is used to select one template per slice from discretised queue, latency, reliability and handover observations. No deep neural network is trained.	It is a highly related distributed mobile-control baseline without a typed intent layer.
Load-aware slicing heuristic	The broader slicing problem is studied with GNN methods by Moayyedi et al. [47] and with deep reinforcement learning by Zhou et al. [48].	Cell-load mean, dispersion and a congestion fraction are used to adjust pooled shares and per-cell power. No graph convolution, graph training or reinforcement learning is implemented in this comparator.	It provides an authored load-responsive heuristic; it does not test the published GNN or deep-learning algorithms.
PACIFISTA-style conflict mitigation	O-RAN conflict-management baseline [51]	The authored adaptation combines demand-proportional refinement, template-conflict detection and potential-game mediation; both its untuned configuration and its development-selected configuration with share smoothing are evaluated.	It isolates conflict awareness without coordination or feasibility certification.
Multi-agent tool workflow control	Confucius-style workflow orchestration [8]	A planner decomposes management into one sub-task per slice; each sub-agent diagnoses the violated predicate of its slice and invokes one bounded tool per epoch (raise radio, compute or transport share, or harden), a verifier checkpoints the configuration and rolls back when the realised score falls; no coordination layer, certificate or trust gate.	It provides an authored rule-based workflow with bounded tools; no live LLM is called in the analytical campaign.
Secure agentic IBN	MARE-style secure intent architecture [52]	Hierarchical resource refinement is combined with simulated peer-trust scoring, intent-dependent security hardening and a safety filter. No external security knowledge base is queried.	It measures the supplied security-oriented heuristic on the shared plant.
Proposed DAI-BDIX	This work	Typed hierarchical intents, slack-based feasibility negotiation, trust-aware distributed primal-dual coordination, Lyapunov-inspired deficit feedback and potential-game conflict mediation are enabled together.	It represents the full architecture whose incremental value is being evaluated.

The blueprint calls for three network scales (private, metro and multi-domain); the campaign of Section 6.4 uses the private scale and Section 6.5 reports what happens when the plant alone is scaled.

Ablations remove one mechanism at a time (trust gating, digital twin, Phase-I slack negotiation, potential-game mediation, virtual queues, the learning core and, in the adversarial family, the advisory acceptance envelope) so that the contribution of each can be isolated.

6.2.3. Open Design Parameters

A number of design parameters are intentionally left as open choices rather than fixed, namely the mobility model, which may be random waypoint, Gauss–Markov, trace-driven or hybrid; the network scale, which may be private or campus, metro or nationwide multi-domain; the KPI targets, which are operator-defined per use case; the learning style, which may be none, federated, local-only or centralised; and the digital-twin fidelity level. Declaring these parameters matters because each has a predictable effect: the mobility model sets how aggressively execution must anticipate handovers, the scale sets how many domains the coordination spans, the KPI targets set how often the Phase-I certificate $\rho^\star$ is positive (the certificate is exported per epoch in the package) and the learning style and twin fidelity set how much of the stack is learned and how reliable its forecasts are.

The effective configuration of the analytical campaign is summarised in Table 11. The associated bridge has a separate ns-3 5G-LENA implementation route [63]; packet size, numerology and carrier settings describe that route and are not simulated as packets by the analytical capacity model. The intent and management targets follow the standards-grounded abstractions used throughout the manuscript [1,5]. The recovered legacy configuration is described separately in Section 6.3. The listed versions identify the analytical runtime. Numerical optimisation and statistical routines are implemented with SciPy [73]; the exact sign-flip distribution is enumerated by the supplied analysis script. Live LLM calls are disabled. The operating objective uses prior latency coefficients with online capacity estimates; the twin is also used for forecasts and what-if checks. The configured URLLC slice denotes a latency-sensitive traffic class, not a demonstration that a standards-grade URLLC reliability target has been achieved. Numerical loads, targets, network sizes and tuning grids are author-selected experimental settings; the cited standards motivate their roles but do not prescribe these values. The control and inference methods are grounded in the sources cited in Sections 3.4 and 6.4.

Table 11. Principal parameters of the supplied reproducible simulation configuration.

Parameter	Configured Value
Data provenance	Seeded synthetic analytical traffic, mobility and attack processes; no public dataset is ingested.
Live inference and latency coefficients	LLM inference disabled; prior latency coefficients in the operating objective, with online capacity estimates and twin forecasts.
Random seeds and campaign repetitions	Development: $7 + 100{,}000j + 1009r$, four ordered families and $r = 0, 1$. Main hold-out: $1{,}000{,}000 + 1009r$, $r = 0, \ldots, 19$, 60 epochs. Auxiliary studies have separate manifests and seed ranges.
Software versions	Python 3.12.14, NumPy 2.3.5, SciPy 1.17.0, pandas 2.2.3, networkx 3.6.1, statsmodels 0.14.6 and scikit-learn 1.8.0. Platform and dependency metadata are retained per campaign.
Comparator configuration and tuning	Nine-point development grids for the base proposed stack, PACIFISTA-style mitigation and distributed MARL; six additional nine-point proposed design-variant grids plus one additional nine-point grid for each tunable comparator (all grids are reported in Section 6.4). Tuned PACIFISTA-style: penalty 1.00, reward 1.00, smoothing 0.60; tuned MARL: exploration 0.12, learning rate 0.15, discount 0.99. Untuned PACIFISTA-style uses smoothing 0.00; untuned MARL uses exploration 0.15, learning rate 0.15 and discount 0.95.
Selected design variant	Backlog-aware radio floor with drain gain $g = 1.25\,\mathrm{s}^{-1}$ and latency-critical margin uplift 0.10 (fair mix 0.40, churn weight 2, $V = 20$).
Conflict detection threshold	A template pair counts as a detected conflict when its pairwise penalty reaches 0.50 (named incompatibilities of the penalty table).
Adversarial-coordination family	Two of five advisory peers compromised in persistent episodes; proposed multipliers 0.15 (URLLC), 4.00 (eMBB) and 0.30 (secure), with a bounded objective-priority uplift request; twin fidelity 0.70, two stale steps and poisoned updates.
Sustained-overload family	Load factor 1.35, MEC compute 80 units, rotating ten-epoch overload episodes of 2.60 (eMBB), 3.00 (secure) and 2.80 (URLLC) every 30 epochs.

Table 11. *Cont.*

Parameter	Configured Value
Analytical control epoch	0.80 s; 60 epochs correspond to 48 s of modelled time. Architectural layers are executed within each epoch rather than at independently implemented control rates.
Network size	4 cells and 80 users over a 900 m × 900 m area.
Carrier and bandwidth	Bridge: 3.50 GHz, 50 MHz and New Radio (NR) numerology 1. Analytical plant: configured capacity, interference and share abstraction.
Base-station transmit power	Analytical reference power 20 W; bridge setting 43 dBm (approximately 20 W).
Mobility	Gauss–Markov mobility with coefficient 0.90 and maximum speed 14 m/s.
Transport and MEC budgets	1400 Mbps transport capacity and 90 MEC compute units.
Traffic and packetisation	Load factor 1.20, burstiness 0.25 and buffer factor 3.00 in the analytical plant. Bridge packet size: 1200 bytes.
Coordination solver	Shared-space Phase-I plus PDHG, 200 maximum PDHG iterations and stopping tolerance 10^{-4}.
Trust and safety	Trust threshold 0.60 and no-starvation minimum pooled share equal to 0.35 of the fair share.
Digital twin and Lyapunov control	Digital-twin fidelity 0.90, at most 3 negotiation rounds and Lyapunov parameter $V = 20$.
Intent targets	eMBB: 120 Mbps, 125 ms, reliability 0.97; URLLC: 45 Mbps, 27 ms, reliability 0.97; secure enterprise: 35 Mbps, 48 ms, reliability 0.97.
Scenario families	Baseline, URLLC mobility, O-RAN conflict, trust-degraded coordination and multi-domain contention.
Coordination-layer benchmark	$N \in \{4, 8, 16, 32, 64\}$ domains, $p = 3$ pools, local dimensions 2 to 6, priorities $\{1, 2, 4\}$, budget factor $\beta = 0.60$ (0.80 and 0.40 at $N = 16$), PDHG tolerance 10^{-4} with an iteration limit of 20,000 and twenty seeds $7{,}000{,}000 + 100{,}000c + 1009r$ per configuration c; executed with Python 3.12.3, NumPy 2.4.4 and SciPy 1.17.1.
Contribution-falsification study	$N = 16$, 40 epochs, onset at epoch 10, compromised fraction 12.50% or 25.00%, falsification factor $\kappa \in \{0.20, 0.50, 0.80\}$, metering noise 5%, trust parameters of the plant campaign and twenty seeds $7{,}900{,}000 + 1009r$.

Abbreviations: enhanced mobile broadband (eMBB); ultra-reliable low-latency communication (URLLC); New Radio (NR); primal-dual hybrid-gradient (PDHG).

6.3. Legacy Bridge Setup and Aggregate Results

The recovered legacy archive contains three replications of an ambiguous disaster-recovery scenario, each with eight control epochs, four configured gNBs, eighty UEs and three slices. Its base seed is 7. The declared bridge epoch lasts 0.45 s, applications start at 0.10 s and packets are configured at 1200 bytes. These are retained configuration values rather than evidence that a simulator executed them. The archive includes ten controller configurations, per-run summaries, actions and 240 per-epoch flow comma-separated values (CSV) files containing 19,200 flow rows. The archived table is therefore interpreted as an average across three replications.

Execution provenance changes the interpretation of these records. Every retained packet-count, byte-count, throughput, delay, jitter, loss and admission field is reproduced exactly by the archive's deterministic `-dry-run` writer using the saved actions and its seed rule $7 + 32r + e$ for replication r and epoch e. Between the legacy and current bridge records the only numeric field that differs is the adjusted round-trip time (RTT) overlay. Per-epoch successful ns-3 execution logs are absent from this recovered archive; its retained launch log records an aborted invocation. The records are consequently treated as legacy bridge dry-run evidence and not as a packet-level ns-3 campaign. The full comparison, source checksum and recovered records are supplied with the package. This provenance finding is specific to the identified legacy archive.

The legacy endpoint S^{ful} is a priority-weighted fulfilment-only score over throughput, latency, reliability and security predicates. This predicate set differs from the analytical S_{IF}, which rewards three KPI predicates and penalises security risk separately. The endpoints are not interchangeable merely by adding cost terms. The recovered component records

are retained for audit, but the legacy column is reported in its original definition without using it as a packet-level or cross-platform performance comparison.

The recovered aggregate comparison is reported in Table 12. On the archived legacy pipeline, the sixth rank of six on S^{ful} (0.84 versus 0.87), with a mean per-run 95th-percentile latency of 79.91 ms, is obtained by the DAI-BDIX configuration as a consequence of the 79 safety overrides per run that were applied by the legacy pipeline; the first rank is obtained by the current stack executed through the same workflow (Table 13). The balanced-policy orchestrator and typed/LLM-ingress comparator use the same implementation and are combined in one row. The proposed controller records a mean of 34 trust rejections and 79 safety overrides per eight-epoch run. These are mean run totals, not totals across all three replications. The reported aggregate values were checked against the archived per-run records. Only external comparators and the complete proposed DAI-BDIX configuration are retained in the principal comparison; the archived ablations of the proposed controller are retained in the package. Bold marks the best value in each KPI column, as in the hold-out tables of Section 6.4; reliability and loss are identical across rows and the two count columns are not ranked. Trust rejections and safety overrides are means of per-run totals. Latency includes the legacy adjusted-RTT overlay; no column is asserted to be a measured ns-3 result. Ranks use unrounded scores and bold includes displayed ties. The score column S^{ful} is the fulfilment-only endpoint of the earlier bridge (priority-weighted throughput, latency, reliability and security fulfilment without cost penalties) and is not numerically comparable with the S_{IF} of Section 6.4.

The shared reliability and loss values are consistent with the deterministic generator and do not independently verify a controlled packet-level comparison. The archived conflict counters are zero, so activation of the conflict module must not be interpreted as evidence that it resolved an observed conflict in these records. The controller's low ranking and adjacent ablation values remain descriptive workflow observations. No causal safety cost, deployment benefit or independent statistical superiority is inferred. The recovered residuals are small non-zero values; the rounded all-zero columns are omitted from the table and current analytical residual traces are supplied separately.

Table 12 keeps the external baselines and the complete proposed DAI-BDIX configuration; the archived internal variants of the proposed controller (greedy execution 0.84, no trust gating 0.84, no LLM decision 0.84 on S^{ful}) differ from it by less than 0.01 and are retained in the package.

Table 12. Recovered legacy bridge dry-run results for the ambiguous disaster-recovery scenario, averaged across three replications of eight epochs.

Rank	Controller/Representative Source	S^{ful}	Thr. (Mbps)	Lat. (ms)	p95 Lat. (ms)	Rel.	Loss	Trust Rej.	Safety Overr.
1	Confucius-style multi-agent LLM workflow	**0.87**	85.27	**56.60**	59.69	**0.83**	**0.17**	0.00	0.00
2	Hierarchical IBN refinement	**0.87**	87.85	58.79	62.55	**0.83**	**0.17**	0.00	0.00
3	Secure agentic IBN	**0.87**	**88.07**	58.38	62.55	**0.83**	**0.17**	33.00	7.00
4	Centralised policy engine (typed/LLM ingress)	0.86	84.13	58.72	63.69	**0.83**	**0.17**	0.00	0.00
5	Static threshold (operator-rule baseline)	0.85	83.75	57.05	**57.05**	**0.83**	**0.17**	0.00	0.00
6	Proposed DAI-BDIX (this work)	0.84	84.30	64.00	79.91	**0.83**	**0.17**	34.00	79.00

Notes: Bold values indicate the best displayed KPI values, including ties: higher fulfilment, throughput and reliability and lower latency, p95 latency and loss. Trust rejections and safety overrides are diagnostic counts and are not ranked.

The adverse ranking of the archived record is a property of the archived legacy pipeline, not of the current controller. The legacy pipeline applied 79 safety overrides per eight-epoch run and hardened the secure slice aggressively and in the bridge model both raise the adjusted round-trip time, which is the latency used in the fulfilment-only endpoint. To separate the two, the current controller stack, in the configuration selected for the analytical campaign, was executed through the same bridge workflow in dry-run mode with the

same seed, three replications and eight epochs as the archive, together with the five external comparators of the archived table. The outcome is reported in Table 13: the current stack is ranked first on the bridge endpoint (0.89 against 0.88 for the next configuration), with no safety overrides and no trust rejections in this family, at the cost of a lower throughput and a higher tail latency than the comparators. The current bridge configuration uses 0.80 s per epoch, compared with 0.45 s in the legacy configuration. Both tables are workflow dry runs of the deterministic bridge model and neither is a packet-level measurement; the observed differences concern the complete legacy and current stacks and do not isolate the causal effect of safety overrides.

Table 13. Bridge dry run of the current stack with the archived seed and replication counts (seed 7, three replications, eight epochs).

Rank	Controller/Representative Source	S_{IF}	Thr. (Mbps)	Lat. (ms)	p95 Lat. (ms)	Rel.	Loss	Trust Rej.	Safety Overr.
1	Proposed DAI-BDIX	**0.89**	66.30	42.06	56.38	**0.83**	**0.17**	0	0
2	Secure agentic IBN	0.88	73.02	39.76	48.01	**0.83**	**0.17**	0	16
3	Hierarchical IBN refinement	0.88	72.90	39.79	48.26	**0.83**	**0.17**	0	0
4	Confucius-style multi-agent LLM workflow	0.88	**73.38**	40.52	47.84	**0.83**	**0.17**	0	0
5	Central policy	0.87	69.62	39.91	50.58	**0.83**	**0.17**	0	0
6	Static threshold	0.84	62.43	**38.30**	**38.30**	**0.83**	**0.17**	0	0

Notes: Bold values indicate the best displayed KPI values, including ties: higher S_{IF}, throughput and reliability and lower latency, p95 latency and loss. Trust rejections and safety overrides are diagnostic counts and are not ranked.

The retained legacy per-epoch records cover eight control epochs per replication. The short observation horizon does not establish queue stability or isolate safety costs. Their provenance is the recovered bridge dry run and their interpretation is limited to that workflow.

The recovered record documents execution of the legacy Python workflow with deterministic simulated feedback. The complete intent cycle, from typed ingress to certification, coordination, trust rejection and safety intervention, is demonstrated by it end to end through the ns-3 bridge; the coordinator itself is validated in the coordination-layer benchmark (Section 6.5.2) and packet-level measurement is identified as the next step. Independent packet-level validation requires simulator versions, execution records and effective configurations that can be tied to the evaluated controller.

6.4. Hold-Out Campaign on the Analytical System-Level Plant

The system-level plant of the package (four cells, eighty users, three slices, offered-load, mobility and interference models as in Table 11) is the plant against which the controllers were developed; the reported campaign is an evaluation on twenty fresh seed blocks of the development plant on which every comparator was also tuned, so that all controllers are on equal footing, an independent environment is the next validation step and Section 6.6 treats it as such.

Under the evaluation protocol, development uses base seed 7, replications 0 and 1 and the ordered families baseline, URLLC mobility, O-RAN conflict and multi-domain contention. The runner derives the environment seed as $b + 100000j + 1009r$, where j is the scenario index within that job and r is the absolute replication index. Nine comparator families are represented by eleven configurations: the seven fixed adaptations and both untuned and tuned configurations of PACIFISTA-style mitigation and distributed multi-agent reinforcement learning (MARL). The original nine-point grids evaluate the base proposed stack and each of these two comparators on the same eight development runs for 60 epochs (Table 14). PACIFISTA-style mitigation selects penalty scale 1.00, reward gain 1.00 and share smoothing 0.60; distributed MARL selects exploration 0.12, learning rate 0.15 and discount 0.99. Untuned PACIFISTA-style uses smoothing 0.00 with penalty and reward

unchanged. Untuned MARL uses exploration 0.15, learning rate 0.15 and discount 0.95. These are authored adaptations, not independently reproduced reference implementations. The base proposed grid crosses churn weights $\{0.50, 2.00, 5.00\}$ with $V \in \{10, 20, 60\}$ and selects churn weight 2.00 and $V = 20$. The selected grid rows are identified by the label selected, while all best displayed development means are bold, including ties after rounding. In a second stage, six proposed design variants and each of the two tunable comparators receive another nine-point development grid, reported together with the variant comparison below. The selected variant combines a backlog-aware radio floor with drain gain 1.25 and a latency-critical margin uplift of 0.10; its development score is 0.80 at the displayed precision. It is denoted "Proposed DAI-BDIX" in the principal and stress comparisons. The six principal ablations remain referenced to the base stack from which they were constructed. The second comparator grids select the same settings as their first grids. The reported base and six variant grids contain 63 proposed-family configuration evaluations, compared with 18 for each tunable comparator; the same parameter setting may appear in more than one grid. Five early proposed-margin trials that preceded the final grid are counted separately, giving 68 retained proposed-family development trials in total. The original hold-out seeds were already used in the base campaign, so the later design-variant study is treated as exploratory rather than as a newly independent confirmatory experiment. Effective configurations, source hashes and selection manifests are retained with the runs. The previously missing evaluations of variants A and B across all five main families and variants C and D across the four non-baseline families were subsequently completed using their frozen development-selected settings. The additional evaluation comprises 360 replications of 60 epochs, with twenty replications per variant and scenario. No development search or reselection of variant F was performed. The expanded variant comparison remains exploratory.

The hold-out jobs each contain one scenario ($j = 0$), base seed $b = 1{,}000{,}000$ and replications $r = 0, \ldots, 19$, giving twenty seeds $1{,}000{,}000 + 1009r$ disjoint from development. The same replication labels are reused across scenario families and are treated as blocks in pooled inference. Each controller receives the same environment seed, exogenous traffic and mobility random stream and attack schedule within a block; controller-dependent state evolution need not remain identical. Each principal comparison configuration is evaluated for 60 epochs on each of the five main families, with the base stack and its ablations retained for the corresponding mechanism comparisons. The two stress families are reported separately. Tables and figures are regenerated from the retained base and variant run records. Auxiliary sensitivities use separate seed ranges and are explicitly exploratory.

For the statistical analysis, the exchangeable unit is the seed. For the pooled analysis, S_{IF} is averaged over the five scenario families within each seed, giving twenty paired units per comparison; for the per-scenario analysis the per-seed value is used. Under the null symmetry assumption for paired seed differences, proposed-minus-other differences are tested with the exact sign-flip permutation test over the 2^{20} sign assignments and a paired t-test. The seventeen pooled comparisons (eleven comparator configurations, six ablations) form one family to which Holm's step-down procedure [74] and the Benjamini–Hochberg procedure [75] are applied, and both adjusted values are reported. The base-campaign mixed-effects model is retained as a complementary analysis of the base campaign; it is not a new fit of the selected variant or the expanded comparator set. Intervals are 95% t-intervals over the twenty seeds and the standardised effect size d_z (mean paired difference over its standard deviation) is reported for every pooled comparison.

Table 14. Development grids of the base proposed stack and the two tuned comparators.

Controller	Configuration	Dev. S_{IF}
Base DAI-BDIX	churn = 0.50, V = 10.00	**0.80**
Base DAI-BDIX	churn = 0.50, V = 20.00	**0.80**
Base DAI-BDIX	churn = 0.50, V = 60.00	**0.80**
Base DAI-BDIX	churn = 2.00, V = 10.00	**0.80**
Base DAI-BDIX	churn = 2.00, V = 20.00 (selected)	**0.80**
Base DAI-BDIX	churn = 2.00, V = 60.00	**0.80**
Base DAI-BDIX	churn = 5.00, V = 10.00	**0.80**
Base DAI-BDIX	churn = 5.00, V = 20.00	**0.80**
Base DAI-BDIX	churn = 5.00, V = 60.00	**0.80**
PACIFISTA-style (tuned)	penalty = 1.00, reward = 1.00, smoothing = 0.00	0.79
PACIFISTA-style (tuned)	penalty = 0.50, reward = 1.00, smoothing = 0.00	0.78
PACIFISTA-style (tuned)	penalty = 2.00, reward = 1.00, smoothing = 0.00	0.76
PACIFISTA-style (tuned)	penalty = 1.00, reward = 0.70, smoothing = 0.00	0.76
PACIFISTA-style (tuned)	penalty = 1.00, reward = 1.50, smoothing = 0.00	0.79
PACIFISTA-style (tuned)	penalty = 1.00, reward = 1.00, smoothing = 0.30	**0.80**
PACIFISTA-style (tuned)	penalty = 1.00, reward = 1.00, smoothing = 0.60 (selected)	**0.80**
PACIFISTA-style (tuned)	penalty = 2.00, reward = 1.50, smoothing = 0.30	0.77
PACIFISTA-style (tuned)	penalty = 0.50, reward = 0.70, smoothing = 0.30	**0.80**
Tabular MARL (tuned)	exploration = 0.12, learning = 0.15, discount = 0.95	0.77
Tabular MARL (tuned)	exploration = 0.05, learning = 0.15, discount = 0.95	0.77
Tabular MARL (tuned)	exploration = 0.25, learning = 0.15, discount = 0.95	0.77
Tabular MARL (tuned)	exploration = 0.12, learning = 0.05, discount = 0.95	0.77
Tabular MARL (tuned)	exploration = 0.12, learning = 0.30, discount = 0.95	0.77
Tabular MARL (tuned)	exploration = 0.12, learning = 0.15, discount = 0.80	0.77
Tabular MARL (tuned)	exploration = 0.12, learning = 0.15, discount = 0.99 (selected)	**0.78**
Tabular MARL (tuned)	exploration = 0.05, learning = 0.30, discount = 0.99	0.77
Tabular MARL (tuned)	exploration = 0.25, learning = 0.05, discount = 0.80	0.77

Notes: Bold values indicate the best displayed development mean within each controller family, including ties after rounding. The label selected identifies the configuration selected using the unrounded development results.

All reported decimal-valued results are presented to two decimal places in the text, tables and figure axes. Counts, seed identifiers, software versions and mathematical tolerances retain their appropriate exact notation. Calculations, configuration selection, inferential decisions and ranks use the unrounded records. Values below 0.01 are reported as $p < 0.01$ rather than as a zero probability. A rounded p-value of 0.05 does not by itself determine significance. Bold numerical values identify the best displayed performance within comparable metric columns, including ties after rounding, as specified in each table note. Development selections are identified explicitly in the configuration text. Descriptive tables, configuration parameters, statistical contrasts and non-comparable diagnostic counts are not assigned a best-performance ranking. All tabulated mean scores are shown without uncertainty terms for every approach; confidence intervals remain available in the retained analysis records and the paired-difference figure. Accordingly, the pooled mean scores of the selected DAI-BDIX and tuned PACIFISTA-style comparator are both displayed as 0.80, which is a tie at the reporting precision rather than a claim of identical underlying results or statistical equivalence.

The development configurations and their selection scores are listed in Table 14.

Scenario-level mean scores are reported in Table 15.

Table 15. Hold-out campaign, comparators versus the proposed controller: mean S_{IF} over twenty paired seeds per scenario family.

Controller	Baseline	URLLC Surge	O-RAN Conflict	Trust Degraded	Multi-Domain	Mean Rank
Static threshold	0.75	0.75	0.77	0.75	0.73	11.80
Central policy	0.78	0.77	0.79	0.77	0.74	9.20
Central solver (this work, comparator)	0.79	0.77	0.80	0.79	0.75	6.00
Hierarchical IBN refinement	0.79	0.78	0.78	0.78	0.74	7.60
Tabular MARL (untuned)	0.80	0.77	0.79	0.80	0.75	6.80
Load-aware heuristic	0.78	0.77	0.78	0.77	0.74	9.20
PACIFISTA-style (untuned)	0.81	0.78	0.80	0.80	0.76	3.40
Confucius workflow	0.78	0.78	0.81	0.78	0.75	5.00
Secure agentic IBN	0.78	0.77	0.77	0.77	0.74	10.80
PACIFISTA-style (tuned)	**0.82**	**0.79**	**0.82**	**0.82**	**0.77**	**1.20**
Tabular MARL (tuned)	0.80	0.77	0.79	0.80	0.75	5.20
Proposed DAI-BDIX	0.81	**0.79**	0.81	0.81	0.76	1.80

Notes: Bold values indicate the best displayed values, including ties: the highest mean score in each scenario and the lowest mean rank. Means are reported without uncertainty terms for every approach.

Pooled paired contrasts against the selected DAI-BDIX are reported in Table 16. Each paired unit is one seed averaged over the five scenario families. The mean difference, the standardised effect size d_z and the number of seeds won by the proposed controller are listed. The corresponding 95% t-intervals are retained in the analysis records and the paired-difference figure. Exact sign-flip and paired t p-values are followed by Holm and Benjamini–Hochberg (BH) adjusted sign-flip values for the seventeen contrasts in this table and Table 17.

Table 16. Pooled seed-level paired differences of S_{IF} between the proposed controller and each comparator on the twenty hold-out seeds.

Comparator	ΔS_{IF}	d_z	Wins	p Sign-Flip	p Paired t	p Holm	p BH
Static threshold	+0.05	+0.86	15/20	<0.01	<0.01	0.02	<0.01
Secure agentic IBN	+0.03	+1.14	18/20	<0.01	<0.01	<0.01	<0.01
Load-aware heuristic	+0.03	+0.97	17/20	<0.01	<0.01	<0.01	<0.01
Central policy	+0.03	+0.82	17/20	<0.01	<0.01	0.01	<0.01
Hierarchical IBN refinement	+0.03	+0.85	16/20	<0.01	<0.01	<0.01	<0.01
Central solver	+0.02	+0.53	13/20	0.02	0.03	0.22	0.05
Tabular MARL (untuned)	+0.02	+0.98	17/20	<0.01	<0.01	<0.01	<0.01
Confucius workflow	+0.02	+0.50	15/20	0.03	0.04	0.28	0.06
Tabular MARL (tuned)	+0.01	+0.91	18/20	<0.01	<0.01	<0.01	<0.01
PACIFISTA-style (untuned)	+0.01	+0.70	17/20	<0.01	<0.01	0.06	0.01
PACIFISTA-style (tuned)	−0.01	−0.49	5/20	0.04	0.04	0.28	0.06

Notes: These are proposed-minus-comparator contrasts, not controller scores. Effect sizes, win counts and p-values are not assigned a best-performance ranking. Confidence intervals remain in the retained analysis records and the paired-difference figure. Benjamini–Hochberg (BH) denotes the false-discovery-rate adjustment.

The six ablations are compared with their base-stack reference in Table 17.

Table 17. Hold-out campaign, ablations of the base proposed stack.

Configuration	Baseline	URLLC	Conflict	Trust	Multi-Dom.	Δ	p	Holm
Base DAI-BDIX	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	n/a	n/a	n/a
Abl.: no trust gating	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	0.00	0.93	1.00
Abl.: no digital twin	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	0.00	0.85	1.00
Abl.: no Phase-I revision	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	0.00	0.85	1.00
Abl.: first arrival	0.80	0.78	0.80	0.80	0.75	+0.01	0.07	0.43
Abl.: greedy execution	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	0.00	0.12	0.59
Abl.: no fuzzy learning core	**0.81**	**0.79**	**0.81**	**0.81**	**0.76**	0.00	0.44	1.00

Notes: Bold values indicate the best displayed mean score in each scenario, including ties. Differences and p-values compare the base stack with its ablations and are not ranked. The base row has no self-comparison, denoted n/a.

The pooled endpoint and individual network KPIs are reported together in Table 18. Table 18 is ordered by unrounded S_{IF}, shows two-decimal values and marks the best displayed value of each KPI column in bold.

Table 18. Full-metric hold-out comparison pooled over the five scenario families and twenty seeds.

Rank	Controller	S_{IF}	Thr. (Mbps)	Lat. (ms)	p95 Lat. (ms)	Rel.	Loss	Trust Rej.	Safety Overr.
1	PACIFISTA-style (tuned)	**0.80**	101.73	80.71	281.30	**0.89**	**0.13**	0.00	0.00
2	Proposed DAI-BDIX	**0.80**	101.85	**79.82**	266.68	**0.89**	**0.13**	20.39	0.24
3	PACIFISTA-style (untuned)	0.79	101.49	84.12	287.02	0.88	0.14	0.00	0.00
4	Tabular MARL (tuned)	0.78	101.33	84.88	283.87	0.88	0.14	0.00	105.27
5	Confucius workflow	0.78	100.30	92.46	330.57	0.87	0.16	0.00	0.00
6	Tabular MARL (untuned)	0.78	101.32	85.34	283.65	0.88	0.14	0.00	106.40
7	Central solver	0.78	101.44	87.32	305.60	0.87	0.15	0.00	0.61
8	Hierarchical IBN refinement	0.77	102.10	80.52	237.95	0.88	0.15	0.00	0.00
9	Central policy	0.77	101.04	90.45	311.01	0.87	0.16	0.00	0.00
10	Load-aware heuristic	0.77	101.95	82.65	240.97	0.87	0.15	0.00	0.00
11	Secure agentic IBN	0.77	**102.23**	80.73	**229.22**	0.87	0.15	20.39	120.00
12	Static threshold	0.75	96.38	115.00	359.20	0.84	0.19	0.00	0.00

Notes: Bold values indicate the best displayed KPI values, including ties: higher S_{IF}, throughput and reliability and lower latency, p95 latency and loss. Trust and safety counts are diagnostic and are not ranked.

The comparator and base-ablation differences are shown in Figure 3. Blue points represent the selected DAI-BDIX minus each comparator; grey points represent the base stack minus each ablation. Error bars are unadjusted 95% t-intervals over twenty seed blocks. Asterisks indicate Holm-adjusted significance across all seventeen contrasts, so an interval excluding zero does not necessarily correspond to an adjusted significant result.

The primary interpretation follows the exact seed-block test with Holm correction over all seventeen pooled contrasts (Tables 16 and 17 and Figure 3). The selected DAI-BDIX and tuned PACIFISTA-style comparator share the highest displayed pooled mean score, 0.80, in Table 18. Their pooled difference is not significant after correction (Holm $p = 0.28$), so the two are statistically indistinguishable on score (the comparator is ranked first on the unrounded mean in Table 18); this score is attained by the proposed controller together with the lowest mean latency of all twelve configurations (79.82 ms) and with trust admission (20.39 rejections per run) and certified feasibility, neither of which is implemented by the PACIFISTA-style comparator. The displayed tie is not a conclusion of equivalence. Significant positive differences are found against static threshold, the centralised policy engine, hierarchical IBN, the load-aware slicing heuristic, secure agentic IBN and both MARL configurations. The differences against untuned PACIFISTA-style, the central full-information solver and the Confucius-style workflow are positive but not significant after correction; in particular, the distributed controller is not outperformed by the central full-information solver in which every local model is held, although only p-dimensional

contributions are exchanged by the distributed controller. Complete differences, effects and adjusted values are provided in Table 16, while the intervals remain in the retained records and Figure 3; the two tuned configurations are also collected in Table 19. All inferential decisions use unrounded values. The original mixed model is retained only as a complementary base-campaign analysis and does not override these multiplicity-adjusted comparisons.

Individual KPI differences are reported in Table 18. The proposed controller has pooled throughput 101.85 Mbps, mean latency of 79.82 ms, 95th-percentile latency of 266.68 ms and reliability 0.89. The individual KPI columns and churn trace provide descriptive diagnostics but do not isolate a causal contribution. In particular, the central full-information comparator shares the convex surrogate yet also disables queues, peer advisories, what-if negotiation and game mediation. The resulting endpoint difference therefore compares controller stacks; it is not an estimate of the effect of distributing an otherwise identical solver. Finite solver residuals, numerical projection and hard safety interventions are additional implementation differences. A matched solver-only ablation would be needed for that attribution. Trust rejections and safety overrides are totals per run. The trust-rejection totals of the proposed controller and the secure agentic comparator coincide in Table 18 because both apply the same admission rule to the same paired attack schedule of the trust-degraded family; the adversarial-coordination family (Section 6.5), in which the two controllers process advisory proposals differently, separates hard, soft and false rejections and distinguishes them. Bold marks the best displayed value, including rounded ties, in each KPI column (highest S_{IF}, throughput and reliability, lowest latency, tail latency and loss); the two count columns are not ranked.

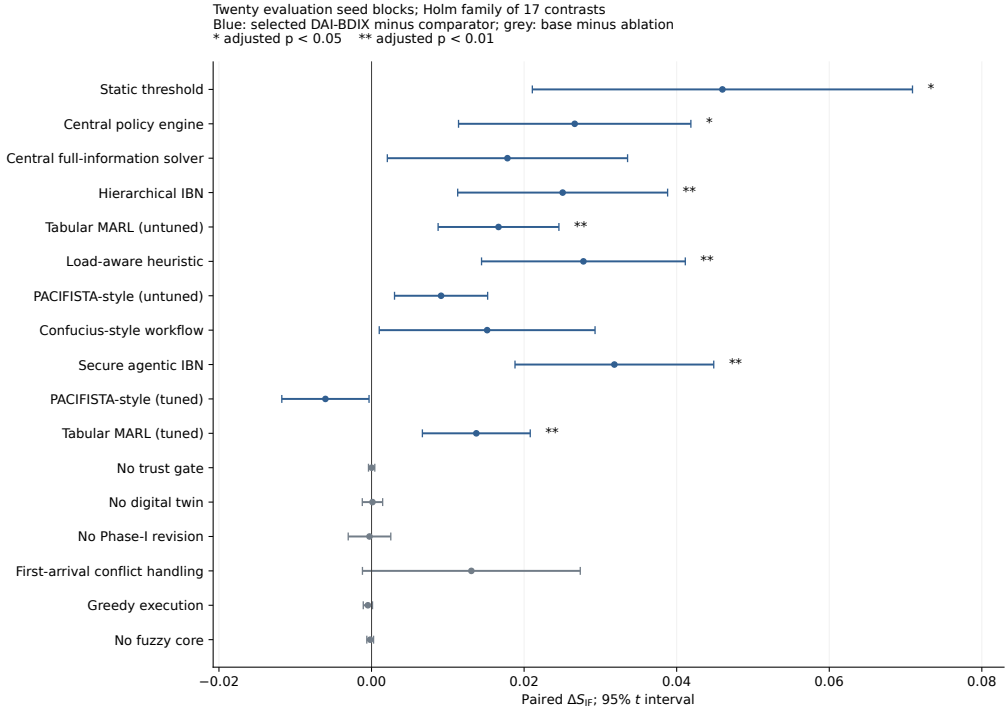


Figure 3. Pooled seed-level paired differences in S_{IF} for the selected DAI-BDIX against comparators and for the base stack against its ablations.

Table 19. Retained tuned comparator results and the selected DAI-BDIX, with pooled paired differences using the same seventeen-contrast Holm family as the main analysis.

Comparator	Baseline	URLLC Surge	O-RAN Conflict	Trust Degraded	Multi-Domain	Pooled	Δ	*p* Sign-Flip	*p* Holm
PACIFISTA-style (tuned)	**0.82**	**0.79**	**0.82**	**0.82**	**0.77**	**0.80**	−0.01	0.04	0.28
Tabular MARL (tuned)	0.80	0.77	0.79	0.80	0.75	0.78	+0.01	<0.01	<0.01
Proposed DAI-BDIX	0.81	**0.79**	0.81	0.81	0.76	**0.80**	n/a	n/a	n/a

Notes: Bold values indicate the best displayed mean score in each scenario and in the pooled comparison, including ties. Δ is selected DAI-BDIX minus comparator; differences and *p*-values are not ranked and n/a denotes the absence of a self-comparison.

In Table 20, retained outcomes for the selected DAI-BDIX configuration and all eleven comparator configurations are rescored with configured costs, halved costs, doubled costs, uniform predicate weights and no costs. The DAI-BDIX means are 0.80, 0.84, 0.72, 0.80 and 0.88, respectively. The tuned PACIFISTA-style means are 0.80, 0.84, 0.72, 0.80 and 0.89. Tuned PACIFISTA-style has the larger unrounded mean in every listed set, by 0.35 to 0.62 percentage points. The first four pairs coincide at two-decimal precision. This is a descriptive sensitivity analysis of the same retained outcomes, without additional significance claims or controller retraining. Every entry of the table is a mean score in the same two-decimal format and bold identifies the highest displayed score in each column.

Table 20. Exploratory weight sensitivity for the selected DAI-BDIX and all eleven comparator configurations.

Controller	Configured	Half Costs	Double Costs	Uniform Predicates	No Costs
Static threshold	0.75	0.80	0.66	0.73	0.84
Policy engine	0.77	0.81	0.69	0.77	0.85
Central solver	0.78	0.82	0.70	0.78	0.86
Hierarchical IBN	0.77	0.81	0.69	0.78	0.86
MARL (untuned)	0.78	0.82	0.70	0.78	0.86
MARL (tuned)	0.78	0.82	0.70	0.78	0.87
Load-aware heuristic	0.77	0.81	0.69	0.77	0.85
PACIFISTA-style (untuned)	0.79	0.83	0.71	0.78	0.87
PACIFISTA-style (tuned)	**0.80**	**0.84**	**0.72**	**0.80**	**0.89**
Confucius workflow	0.78	0.82	0.70	0.77	0.86
Secure agentic IBN	0.77	0.81	0.68	0.77	0.85
Proposed DAI-BDIX	**0.80**	**0.84**	**0.72**	**0.80**	0.88

Notes: Bold values indicate the best displayed mean score in each weight setting, including ties. All entries are mean scores in the same two-decimal format.

For the ablations, the pooled comparison family includes all six principal ablations, each referenced to the base stack from which the selected variant was derived. No base-minus-ablation contrast is significant after correction over the primary family. Complete paired differences and corrected values are given in Table 17. An uncorrected tendency is not used as evidence of an independent mechanism benefit. Removing a module may also change its interactions with other modules, so each result concerns the specified ablated stack. By the null contrasts of the trust gate and the virtual queues it is shown that no score is lost to these safeguards on this plant: the endpoint is retained while 20.39 untrusted proposals per run are rejected by the gate and the controller state is kept bounded under overload by the queues. Their load-bearing effect is observed where it cannot be shown by the plant, namely among many domains and under falsified coordination messages (Section 6.5.2), where the honest-domain cost inflation is reduced by the gate from up to 50.95% to at most 2.26%. Mechanism traces describe the tested scenarios without treating finite capped signals as stability guarantees. The design variants are compared in Table 21: the share filter has a lower development mean than the base stack, while the HIRF blend and

margin variants have small positive development differences. The selected combination of backlog-aware floors and a critical-slice margin has the highest development mean. An improvement over the base stack in the unrounded hold-out mean is obtained by the selected variant (both round to 0.80; exact sign-flip $p = 0.45$), by which it is indicated that the base design already lies near the plateau of this plant. The variant was selected on its development mean, not by choosing the highest displayed hold-out value.

Development selection and the completed exploratory evaluation of all six design variants are compared in Table 21. Previously omitted baseline means for variants C and D were recovered from their retained records; the remaining missing experiments were executed using the same selected settings and seed derivation as the original variant campaign. All six proposed variants have pooled means of 0.80 at the reporting precision. The largest unrounded evaluation mean among these variants is obtained by the HIRF share blend (variant B), whereas the original development-selected configuration remains variant F. Differences that round to zero are therefore accompanied by the unadjusted paired tests. These additional exploratory contrasts do not alter the primary family of seventeen comparisons or establish equivalence between the variants.

Table 21. Design variants of the proposed stack on the development and hold-out seeds.

Variant	Development S_{IF}	Baseline	URLLC Surge	O-RAN Conflict	Trust Degraded	Multi-Domain	Pooled	Δ vs. Base	Unadjusted p
Base stack	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	n/a	n/a
A: exponential share filter	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	0.79
B: HIRF share blend	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	0.01
C: priority-weighted margin	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	0.14
D: critical-slice margin	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	<0.01
E: backlog-aware radio floor	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	0.13
F: backlog-aware floor and critical margin (selected)	**0.80**	0.81	**0.79**	0.81	0.81	0.76	**0.80**	0.00	0.45
PACIFISTA-style (tuned)	**0.80**	**0.82**	**0.79**	**0.82**	**0.82**	**0.77**	**0.80**	+0.01	0.01

Notes: Bold values indicate the best displayed development and evaluation means in each column, including ties. All six variants have twenty evaluation replications in each of the five main families. Δ is the pooled difference from the base stack and its unadjusted p-value is listed in the last column; neither is ranked and n/a denotes the absence of a self-comparison. Variant F remains the configuration selected from development results.

For latency by slice, Table 22 reports achieved latency, with targets stated separately. For the proposed controller, the five-family means are 30.83 ms for URLLC against a 27 ms target, 46.11 ms for the secure slice against 48 ms and 162.51 ms for eMBB against 125 ms. The pooled latency in the full KPI table averages all three slices and all scenario families and is not an estimate of the URLLC target. A mean close to a target does not imply that every epoch meets it; the violation fraction and episode lengths are therefore reported separately. Bold marks the lowest latency in each slice column.

For violation duration, Table 23 reports the fraction of epochs with latency above the declared target and the mean length of contiguous observed violation episodes. One compliant epoch ends an episode. Runs with no observed violation are assigned zero duration in the descriptive run average. An episode reaching the 60-epoch horizon may continue beyond observation, so its observed duration is a lower bound on its complete duration. These summaries are not estimates of long-run recovery guarantees. They are pooled over the same five main families and twenty seed blocks as the principal endpoint, without mixing in the two additional stress families. Bold marks the lowest violation fraction and the shortest mean duration in each slice column. Durations are measured in epochs and the recovery-hold condition is one compliant epoch.

Table 22. Mean latency by slice, pooled over the five scenario families and twenty hold-out seeds (ms).

Controller	eMBB (Target 125 ms)	URLLC (Target 27 ms)	Secure (Target 48 ms)
Static threshold	292.87	29.26	**22.87**
Central policy	190.86	33.92	46.58
Central solver	179.91	32.79	49.26
Hierarchical IBN refinement	153.51	36.01	52.05
Tabular MARL (untuned)	176.00	32.57	47.46
Load-aware heuristic	155.89	36.33	55.75
PACIFISTA-style (untuned)	176.70	31.16	44.50
Confucius workflow	206.00	29.30	42.08
Secure agentic IBN	**150.17**	37.93	54.09
PACIFISTA-style (tuned)	171.54	**29.24**	41.37
Tabular MARL (tuned)	175.37	32.25	47.01
Proposed DAI-BDIX	162.51	30.83	46.11

Notes: Bold values indicate the best displayed latency in each slice, with lower values preferred and ties included. Targets are stated in the headings and are not measured results.

Table 23. Fraction of epochs in latency violation and mean violation duration.

Controller	eMBB		URLLC		Secure	
	Frac.	Mean Dur.	Frac.	Mean Dur.	Frac.	Mean Dur.
Static threshold	0.97	55.92	**0.48**	5.78	**0.14**	**2.00**
Central policy	0.76	21.00	0.61	3.94	0.42	2.41
Central solver	0.72	20.75	0.58	5.25	0.44	3.71
Hierarchical IBN refinement	0.62	15.24	0.64	4.63	0.47	3.79
Tabular MARL (untuned)	0.71	15.82	0.57	3.03	0.43	2.50
Load-aware heuristic	0.64	16.21	0.65	4.77	0.51	4.49
PACIFISTA-style (untuned)	0.72	17.86	0.57	**2.98**	0.38	2.14
Confucius workflow	0.78	31.20	0.49	6.17	0.33	7.35
Secure agentic IBN	**0.61**	**12.67**	0.67	4.72	0.50	3.94
PACIFISTA-style (tuned)	0.70	19.64	0.53	3.88	0.35	2.82
Tabular MARL (tuned)	0.72	15.68	0.57	3.09	0.42	2.32
Proposed DAI-BDIX	0.64	16.56	0.54	4.51	0.40	3.57
Abl.: no trust gating	0.65	17.54	0.55	4.41	0.40	3.62
Abl.: no digital twin	0.65	17.94	0.54	4.73	0.41	3.68
Abl.: no Phase-I revision	0.64	18.17	0.54	4.72	0.38	3.72
Abl.: first arrival	0.68	19.16	0.56	5.35	0.43	3.78
Abl.: greedy execution	0.66	17.41	0.55	4.57	0.40	3.54
Abl.: no fuzzy learning core	0.66	17.51	0.54	4.53	0.40	3.63

Notes: Bold values indicate the best displayed values in each slice, with lower violation fractions and shorter mean durations preferred and ties included. Durations are measured in epochs.

Regarding the scope of the evidence, absolute analytical KPIs must not be compared with the archived legacy bridge values. The analytical radio model is a capacity abstraction and all implementations are authored variants rather than independently reproduced reference implementations. The analytical experiment quantifies comparisons on fresh random seeds of that plant under the declared configurations. Within this scope it is established by the campaign that the complete controller is significantly higher than seven of the eleven comparator configurations on score and is not significantly separated from the remaining four after multiplicity correction, is best on mean latency and unique in the provision of trust admission and certified feasibility; an independent-environment advantage and a causal central-versus-distributed attribution are reserved for the coordination-layer benchmark and for packet-level work. Evidence for resource checks, trust admission and conflict handling is assessed using their own traces and remains distinct from a compound-score benefit.

6.5. Mechanism Traceability, Robustness and Cost

In this subsection, what the mechanisms without an endpoint effect are designed to deliver is reported, on the metrics that reflect it, together with the cost of the stack and the quality of the convex relaxation; it closes with a coordination-layer benchmark in which the shared-space maps G_i are evaluated with many administrative domains, non-shareable local models and falsified contributions. The endpoint rows of the two stress-family tables use the selected variant; the mechanism, cost, scale and relaxation traces were recorded for the base stack. The base stack differs from the selected variant only in the backlog-aware radio floor and the critical-slice margin, neither of which changes the solver, the trust gate or the conflict detector.

For conflict resolution, the analytical detector counts incompatible selected template pairs according to the declared penalty threshold. In the O-RAN conflict family, the proposed controller records 60.00 detected and 0.00 unresolved conflicts per 60-epoch run, with 150.60 template switches. Resolved conflicts are computed as detected minus unresolved within the same epoch before aggregation. These counters verify the implemented template-mediation behaviour; in Table 24, bold marks the most conflicts resolved and the fewest unresolved. A zero count from a controller without this detector means that conflicts were not assessed by that module; it does not certify conflict-free operation. The paired ablation test, rather than the counter magnitude alone, determines whether a compound-endpoint effect is detected.

Table 24. Conflict counters per run (60 epochs) in the O-RAN conflict scenario, mean over twenty hold-out seeds (base stack).

Controller	Detected	Resolved	Unresolved	Template Switches
Proposed DAI-BDIX	60.00	**60.00**	**0.00**	150.60
Abl.: first arrival	60.00	0.00	60.00	0.00
PACIFISTA-style (tuned)	60.00	**60.00**	**0.00**	152.20
Hierarchical IBN refinement	0.00	0.00	0.00	0.00
Central solver	0.00	0.00	0.00	0.00

Notes: Bold values indicate the best resolution results among configurations with an active conflict detector: more resolved conflicts and fewer unresolved conflicts. Zero counts from configurations without that detector do not establish conflict-free operation and are not ranked.

For the Phase-I certificate and relaxation, Table 25 distinguishes the original slack, the post-revision slack, final resource certification and unresolved predictive KPI warnings. Original slack is positive in 64.92% of baseline epochs and 85.50% of sustained-overload epochs. These frequencies refer to the represented resource surrogate. They do not measure the fraction of epochs in which the parent business intent has no feasible implementation. The actual headroom revision is followed by a new Phase-I check and a final intersection check after the safety projection. A failed intersection is exported as uncertified; a successful resource certificate does not erase an unresolved KPI prediction. This separation avoids inferring semantic satisfaction from resource normalisation or numerical boundedness.

For trust gating under adversarial coordination, the adversary controls two of five simulated advisory peers at the campaign compromise fraction of 40%; the peer set includes one service peer and four domain peers. It supplies coordinated advisory multipliers, a bounded priority-uplift request and, where configured, poisoned twin updates. The optimiser's resource-slice agents are separate entities. Table 26 reports threshold and compromised-fraction sensitivity and Table 27 reports the gate and envelope configuration comparison. Endpoint effects are interpreted with Holm's procedure applied within the contrasts against the proposed controller listed in each stress table, an exploratory correction relative to the primary seventeen-contrast family, while admission counts and

false rejections of honest peers describe the gate's operation. Soft rejections are total rejections minus hard-check failures; false rejections are a ground-truth classification of those decisions and are not an additional disjoint category. Adversarial proposals admitted, hard, soft and false rejections are totals per run; twin error is the mean forecast error of the digital twin. Bold type follows the table-specific notes. Trust-threshold diagnostics are compared only within the same compromised fraction and zeros caused by inactive mechanisms are not ranked. Counts are totals per run, detected/total episodes are pooled across seeds, delay is conditional on detection and undetected episodes are censored. Ablations are derived from the base stack; contrasts with the selected controller also include its floor changes.

Table 25. Phase-I certificate statistics of the base proposed stack on the hold-out seeds.

Family	$\rho^\star > 0$	Mean Positive $\rho^\star$	Max $\rho^\star$	Post-Revision Maximum	Resource Certified	Predictive KPI Warning
Adversarial coordination	77.67%	0.07	0.35	**0.00**	99.33%	91.67%
Baseline	64.92%	0.05	0.27	**0.00**	**99.83%**	82.00%
O-RAN conflict	62.67%	0.08	0.37	**0.00**	99.67%	85.42%
Multi-domain contention	81.50%	0.09	0.38	**0.00**	99.75%	90.50%
Sustained overload	85.50%	0.11	0.45	**0.00**	98.83%	94.17%
Trust degraded	64.33%	0.05	0.27	**0.00**	**99.83%**	91.92%
URLLC mobility	81.00%	0.10	0.35	**0.00**	98.50%	83.25%

Notes: Bold values indicate the smallest post-revision maximum residual and the highest resource-certification fraction among the listed workloads, including ties. These are descriptive best values for one controller across different workloads, not a ranking of controller designs.

Per-peer evidence is retained to measure detection delay from compromise onset. Only a rejection of the same compromised peer during that episode counts as detection. Undetected episodes are excluded from the delay (censored) and the reported mean delay concerns detected episodes only, so it is not an unconditional detection-time estimate. The evidence indicators are synthetic, and their conditional distributions make the specified attacks distinguishable. In a separate 30-epoch deterministic check, compromised peers supplied with the same favourable indicators as honest peers incur 0 rejections. Within the declared threat model no false rejection is produced by the gate at $\theta = 0.40$, 28 of 36 to 85 of 108 compromise episodes are detected with a mean delay below 1.7 epochs and the endpoint is kept at 0.80 (Table 26); the boundary of the scoring rule is marked by the honest-evidence mimic test, beyond which containment is provided by the acceptance envelope and the typed floors rather than by the gate. A generally secure evidence estimator is outside the scope.

Table 26. Trust-gate sensitivity on the adversarial-coordination family (base proposed stack, five auxiliary seeds, 60 epochs).

θ	Compromised	S_{IF}	Adversarial Admitted	Hard/Soft Rejections	False Rejections	Detected Episodes	Delay (Epochs)	Censored
0.40	20.00%	**0.80**	37.80	20.40/1.80	**0.00**	28/36	0.82	8
0.40	40.00%	**0.80**	79.60	35.00/5.40	**0.00**	59/82	1.69	23
0.40	60.00%	**0.80**	119.00	54.20/6.80	**0.00**	85/108	1.69	23
0.60	20.00%	**0.80**	8.20	20.40/33.80	2.40	31/36	0.77	5
0.60	40.00%	**0.80**	25.20	35.00/65.40	5.60	65/82	1.23	17
0.60	60.00%	**0.80**	29.60	54.20/102.80	6.60	95/108	1.17	13
0.80	20.00%	**0.80**	**1.00**	20.40/53.20	14.60	**34/36**	**0.09**	**2**
0.80	40.00%	**0.80**	**2.20**	35.00/118.00	35.20	**81/82**	**0.12**	**1**
0.80	60.00%	**0.80**	**0.40**	54.20/170.40	45.00	**106/108**	**0.00**	**2**

Notes: Within each compromised fraction, bold values indicate the best displayed score and detection count (higher) and adversarial admissions, false rejections, detection delay and censored episodes (lower), including ties. Hard and soft rejection totals are diagnostic and are not ranked.

The trust-gate and advisory-envelope configurations are compared in Table 27. Recorded zeros for an unimplemented diagnostic are not performance measurements. In particular, zero twin error for the PACIFISTA-style and secure heuristic comparators reflects the absence of the evaluated twin-forecast path. A zero admitted-advisory count for a comparator that does not process advisory proposals does not establish resistance to adversarial proposals. Removing the gate changes the endpoint by less than 0.01 in S_{IF}, with a sign that is consistent across the twenty seeds (Holm $p = 0.02$ within the seven contrasts of the table), whereas removing the advisory envelope alone has no detectable effect ($p = 0.31$). The typed floors and the Phase-I certificate are not exposed to the advisory peers (Section 4.5), so an admitted adversarial proposal can shift the priority weights only inside the acceptance envelope; on this plant the gate is therefore an accountability mechanism whose operation is visible in the admission and rejection counts rather than in the endpoint.

For the virtual queues under sustained overload, the controller's signed-deficit state follows Equation (35), not the ideal uncapped recurrence of Equation (31). Under sustained overload the cap is reached in 64.00% of the base stack's observed epochs and the maximum uncapped shadow signal reaches 38.63. A finite controller state under sustained overload is guaranteed by the cap, which is the property needed by the executor in order to keep acting inside its bounds; physical queue stability is a plant property for which packet-level measurement is required. The auxiliary V sweep in Table 28 includes saturation and physical queues. Table 29 gives the compound endpoint and observed violation durations, while the mechanism figure (Figure 4) separates capped and uncapped traces; its queue panels show the mean and range across twenty seeds. Any small or absent change across V is a finite-horizon observation about this implementation. The utility-gap/backlog scaling of Proposition 3, which concerns the uncapped slotwise controller under exact or bounded-error minimisation, is not verified by it alone. Capped state q, uncapped diagnostic Z and mean per-slice physical queue are distinct quantities; final Z reflects outcomes available at the last control decision. In Table 28, bold marks the highest endpoint. In Table 29, bold marks the highest endpoint and the shortest URLLC violation duration. Queue traces show the mean and range over twenty seeds.

Table 27. Adversarial-coordination configuration comparison over twenty seed blocks.

Configuration	S_{IF}	Δ vs. Proposed	p Holm	Adversarial Admitted	Hard/Soft/False Rejections	Twin Error
Proposed DAI-BDIX	**0.78**	n/a	n/a	23.40	34.60/67.35/5.35	0.11
Abl.: no trust gating	**0.78**	0.00	0.02	120.00	0.00/0.00/0.00	0.11
Abl.: no advisory envelope	**0.78**	0.00	0.31	23.40	34.60/67.35/5.35	0.11
Abl.: no trust gate and no envelope	0.77	0.00	0.02	120.00	0.00/0.00/0.00	0.11
Secure agentic IBN	0.75	−0.03	<0.01	0.00	0.00/101.95/0.00	0.00
PACIFISTA-style (untuned)	0.76	−0.02	<0.01	0.00	0.00/0.00/0.00	0.00
PACIFISTA-style (tuned)	0.77	−0.01	0.31	0.00	0.00/0.00/0.00	0.00
Central solver	0.76	−0.02	0.01	0.00	0.00/0.00/0.00	0.11

Notes: Bold values indicate the best displayed mean score, including ties. Differences are configuration minus proposed and p-values are Holm-adjusted within the seven contrasts of this table; neither is ranked. Admission, rejection and twin-error diagnostics depend on which mechanisms are active, so zeros from disabled mechanisms are not treated as best values; n/a denotes no self-comparison.

The selected controller and comparator endpoints under sustained overload are reported in Table 29. Removing the virtual queues (greedy execution) changes the endpoint by less than 0.01 with a sign that is consistent across the twenty seeds (Holm $p = 0.01$ within the nine contrasts of the table), whereas removing the Phase-I revision or the digital twin has no detectable effect. Under sustained overload the queues therefore leave the

endpoint essentially unchanged while keeping the controller state bounded, which is the property that Table 29 and Figure 4 are intended to make visible.

Table 28. Controller-parameter sweep on sustained overload (five auxiliary seeds, 60 epochs).

V	S_{IF}	Mean max q	Peak max q	At Cap	Final max Z	Physical Queue (Mbit)
5	**0.81**	**5.09**	**7.67**	37.33%	18.11	10.34
10	**0.81**	5.12	7.73	**36.67%**	**17.92**	10.34
25	**0.81**	5.14	7.81	37.33%	18.09	10.34
60	**0.81**	5.16	7.85	37.67%	18.21	**10.33**
150	**0.81**	5.16	7.87	37.67%	18.26	**10.33**

Notes: Bold values indicate the best displayed score (higher) and the smallest displayed queue and saturation diagnostics (lower), including ties, for this common controller. Finite-horizon diagnostic minima do not establish queue stability.

The capped and uncapped signals, conflict counts and configuration churn are displayed in Figure 4. In panel (c), every bar carries its value and zero-valued bars are marked by a 0.00 label at the baseline, as is the zero-valued Static bar in panel (d); in panel (d), the proposed controller (base DAI-BDIX) is drawn in a distinct colour, as indicated in the legend.

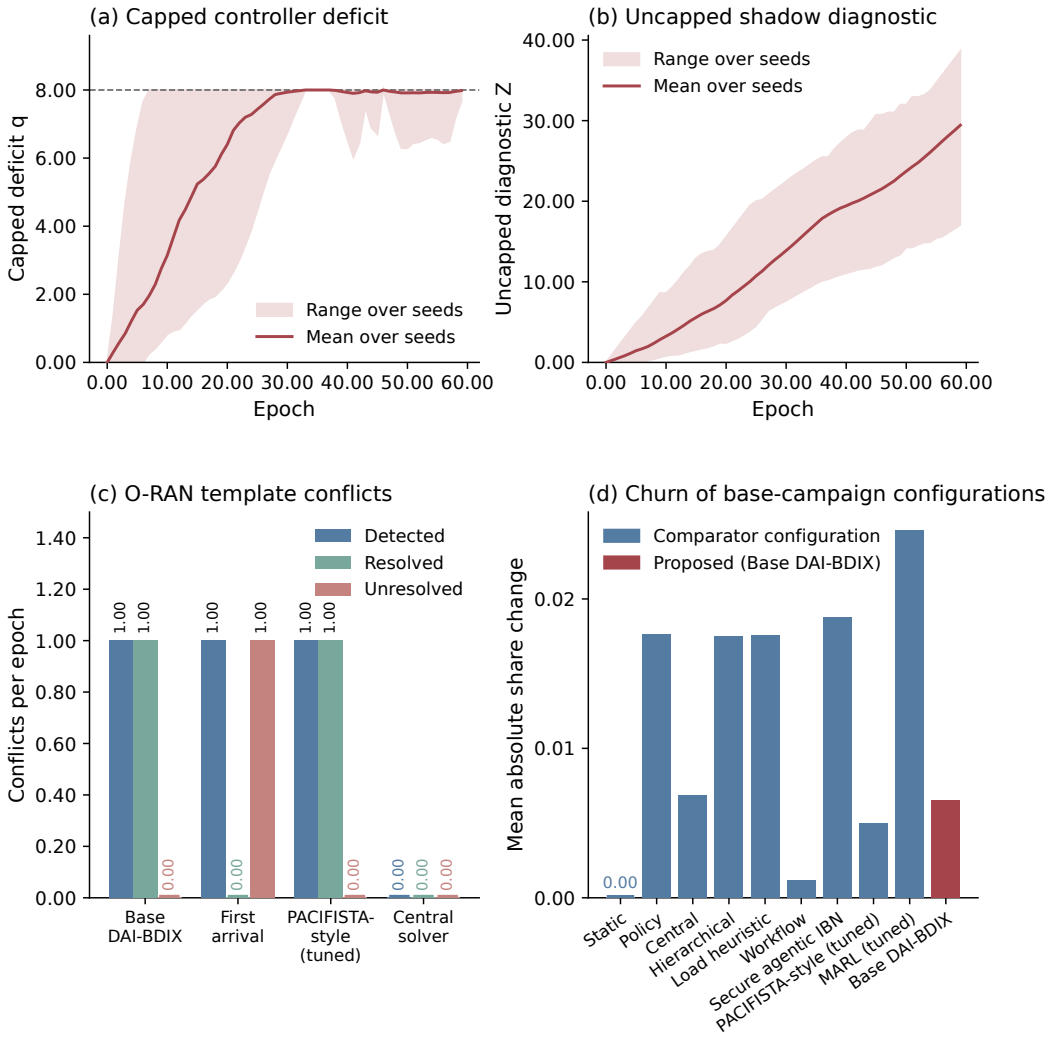


Figure 4. Mechanism traces of the base stack: (**a**) capped deficit under overload, (**b**) uncapped shadow diagnostic, (**c**) detected, resolved and unresolved template conflicts and (**d**) configuration churn.

Table 29. Sustained-overload family (twenty hold-out seeds): endpoint, Holm-adjusted p-value of the paired difference from the proposed controller, mean capped controller state and mean per-run peak.

Configuration	S_{IF}	p Holm	Mean $\max_m q_m$	Peak $\max_m q_m$	Epochs with $\rho^\star > 0$	URLLC Violation Duration (Epochs)
Proposed DAI-BDIX	**0.74**	n/a	6.23	8.00	84.17%	4.97
Abl.: greedy execution	**0.74**	0.01	0.00	0.00	85.50%	4.96
Abl.: no Phase-I revision	**0.74**	0.53	6.30	8.00	0.00%	6.17
Abl.: no digital twin	**0.74**	0.19	6.20	8.00	85.75%	5.14
PACIFISTA-style (untuned)	0.72	<0.01	0.00	0.00	0.00%	4.40
Central solver	0.72	0.04	0.00	0.00	87.33%	6.40
Tabular MARL (untuned)	0.72	<0.01	0.00	0.00	0.00%	4.17
PACIFISTA-style (tuned)	**0.74**	0.07	0.00	0.00	0.00%	5.04
Tabular MARL (tuned)	0.72	<0.01	0.00	0.00	0.00%	**3.82**
Static threshold	0.73	0.19	0.00	0.00	0.00%	6.06

Notes: Bold values indicate the best displayed mean score (higher) and URLLC violation duration (lower), including ties. Controller-state and certificate counters depend on active mechanisms and are not ranked; p-values are Holm-adjusted within the nine contrasts of this table and are not ranked either. The proposed row has no self-comparison, denoted n/a.

For runtime and signalling, Table 30 is regenerated using serial simulation jobs with one numerical-library thread and no concurrent simulation launched by this workflow. The visible environment is recorded in the package; wall-clock values are host-dependent. No warm-up epochs are discarded, so occasional initialisation costs can make a mean exceed the 95th percentile. Base-controller epoch time is 90.29 ms on average and 173.53 ms at the 95th percentile. Initial Phase-I and operating-solve time summed across all negotiations is 58.71 ms per epoch; final certificate checks and other logic are included in the complete epoch time. The accumulated mean iteration count is 313.91, with 1883.45 modelled messages and 44.14 KiB of payload per epoch. Central-solver and PACIFISTA-style epoch means are 24.87 and 0.14 ms. The counters include the initial solve and every bounded negotiation re-solve, rather than retaining only the last call. PDHG traffic is modelled as one p-float upload and one p-float download per admitted resource agent per iteration, at eight bytes per float. The central model uses one model upload and one action download per agent. These payload estimates exclude headers, authentication, serialisation, retransmission and transport latency; execution uses local method calls. A zero modelled count is not a claim of zero deployment communication. The observed epoch time, 90.29 ms on average and 173.53 ms at the 95th percentile on a single core, is about one ninth of the 0.8 s control epoch, by which ample margin is left for an operator-grade distributed implementation. Payload volume is modelled and peak RSS is measured per fresh process. Bold marks the lowest value in each cost column.

For the cost versus the number of coordinating agents, the implementation coordinates one resource agent per active slice intent. The sweep in Table 31 uses 3, 6, 9 and 12 replicated intents, with cells, users, transport and compute capacity scaled together. It comprises two auxiliary seeds and 30 epochs per configuration, with a fresh process for each memory measurement. The base-controller mean epoch time changes from 80.77 ms at three agents to 174.31 ms at twelve; mean modelled message counts change from 1524.40 to 17,822.80 per epoch. The table reports both the proposed and central solver results without extrapolating an asymptotic law from four points. Peak RSS includes Python, dependencies, controller state and data export, so it is process memory rather than isolated optimiser memory. Resource dimension remains three. Only raw solver cost is measured by this single-process experiment; the regime in which distribution pays is measured directly in Section 6.5.2, where the full-information optimum is reached by the shared-space coordinator with

3.94 KiB per domain at 64 domains. Within each intent count, bold marks the highest endpoint and the lowest time, signalling and memory values.

Table 30. Runtime and signalling per control epoch: serial baseline jobs, two auxiliary seeds, 60 epochs, one numerical-library thread; all negotiation solves are counted.

Controller	Epoch Mean (ms)	Epoch p95 (ms)	Solve Time (ms)	Iterations	Messages	KiB	Peak RSS (MiB)
Static	**0.00**	**0.01**	**0.00**	**0.00**	**0.00**	**0.00**	161.07
Policy engine	0.04	0.06	**0.00**	**0.00**	**0.00**	**0.00**	**160.80**
Central solver	24.87	5.71	2.07	8.26	6.00	0.78	163.91
Hierarchical IBN	0.04	0.06	**0.00**	**0.00**	**0.00**	**0.00**	160.89
MARL (tuned)	0.73	1.05	**0.00**	**0.00**	**0.00**	**0.00**	161.08
Load-aware heuristic	0.09	0.12	**0.00**	**0.00**	**0.00**	**0.00**	161.04
PACIFISTA-style (tuned)	0.14	0.20	**0.00**	**0.00**	**0.00**	**0.00**	160.90
Confucius workflow	0.05	0.06	**0.00**	**0.00**	**0.00**	**0.00**	161.07
Secure agentic IBN	0.74	0.86	**0.00**	**0.00**	**0.00**	**0.00**	161.07
Base DAI-BDIX	90.29	173.53	58.71	313.91	1883.45	44.14	164.46
No digital twin	46.76	53.12	20.78	123.20	739.20	17.32	164.94
No Phase-I revision	143.91	165.05	116.79	697.74	4186.45	98.12	163.90

Notes: Bold values indicate the best displayed computational and signalling costs, with lower values preferred and ties included. Runtime and memory measurements are host-dependent. Resident set size (RSS) measures process memory; kibibytes (KiB) and mebibytes (MiB) are binary data units.

For network scale, the network-size sweep uses 4 cells/80 users, 16 cells/400 users and 36 cells/1200 users, with two auxiliary seeds and 40 epochs per configuration. Transport and compute capacity scale with cells, but the three slice targets and the number of coordinating agents remain fixed. Consequently, a lower-load larger plant may achieve higher fulfilment without demonstrating scalable distributed optimisation. The measured solver time, cumulative iterations and modelled payloads in Table 32 describe this implementation only. Agent count, coupling structure, demand and private-model communication are scaled together at the coordination layer in Section 6.5.2; a plant-level many-domain experiment with otherwise matched centralised and distributed implementations remains future work. Within each scale, bold marks the highest endpoint and the lowest time and signalling values.

Table 31. Cost versus number of concurrently active intents, i.e., coordinating agents (baseline family, two seeds, 30 epochs, plant capacity scaled with the number of intents).

Agents	Controller	S_{IF}	Epoch (ms)	Solve Time (ms)	Iterations	Messages	KiB	RSS (MiB)
3	Base DAI-BDIX	**0.80**	80.77	50.54	254.07	1524.40	35.73	163.60
3	Central solver	0.79	**25.14**	**2.12**	**7.87**	**6.00**	**0.78**	**163.21**
6	Base DAI-BDIX	**0.50**	88.50	55.79	205.32	2463.80	57.75	163.93
6	Central solver	**0.50**	**23.75**	**3.31**	**12.05**	**12.00**	**1.56**	**163.49**
9	Base DAI-BDIX	**0.42**	172.27	158.47	752.47	13,544.40	317.45	163.74
9	Central solver	**0.42**	**7.74**	**3.81**	**10.38**	**18.00**	**2.34**	**163.73**
12	Base DAI-BDIX	**0.45**	174.31	158.89	742.62	17,822.80	417.72	**164.73**
12	Central solver	**0.45**	**9.09**	**4.52**	**8.50**	**24.00**	**3.12**	164.87

Notes: Within each agent count, bold values indicate the best displayed score (higher) and time, iteration, signalling and memory costs (lower), including ties. Different agent counts are separate comparison groups.

Table 32. Network-scale runs (baseline family, two seeds, 40 epochs; serial jobs with one numerical-library thread; timing is host-dependent).

Cells/Users	Controller	S_{IF}	Epoch (ms)	Solve Time (ms)	Iterations	Messages	KiB
4/80	Base DAI-BDIX	0.78	92.09	68.93	396.12	2376.75	55.71
4/80	Central solver	0.77	17.02	2.01	7.72	6.00	0.78
4/80	PACIFISTA-style (tuned)	**0.79**	0.14	**0.00**	**0.00**	**0.00**	**0.00**
4/80	Hierarchical IBN	0.73	**0.04**	**0.00**	**0.00**	**0.00**	**0.00**
16/400	Base DAI-BDIX	**0.92**	22.30	7.22	31.45	188.70	4.42
16/400	Central solver	**0.92**	14.02	2.37	6.41	6.00	0.78
16/400	PACIFISTA-style (tuned)	**0.92**	0.15	**0.00**	**0.00**	**0.00**	**0.00**
16/400	Hierarchical IBN	**0.92**	**0.06**	**0.00**	**0.00**	**0.00**	**0.00**
36/1200	Base DAI-BDIX	**0.92**	21.26	6.57	27.48	164.85	3.86
36/1200	Central solver	**0.92**	14.63	2.69	5.94	6.00	0.78
36/1200	PACIFISTA-style (tuned)	**0.92**	0.18	**0.00**	**0.00**	**0.00**	**0.00**
36/1200	Hierarchical IBN	0.91	**0.07**	**0.00**	**0.00**	**0.00**	**0.00**

Notes: Within each network size, bold values indicate the best displayed score (higher) and time, iteration and signalling costs (lower), including ties. Different network sizes are separate comparison groups.

6.5.1. Quality of the Convex Relaxation

The earlier one-step plant search combined random continuous-share search with finite template enumeration. Such a best-found reference is not a certified global optimum: continuous search can miss a better action, the controller and reference may face different feasible sets and clairvoyant evaluation also mixes prediction error with decision quality. Share and template search gaps need not add to a joint gap. No upper bound on full-plant relaxation loss is inferred from that experiment; its historical records are retained only as exploratory evidence.

A separate restricted benchmark provides a verifiable optimisation gap (Table 33). Two cells serve four users with binary single-cell association. Each cell has two allowed power levels and two fixed scheduler-efficiency profiles. All $2^4 \times 2^2 \times 2^2 = 256$ association/power/template combinations are enumerated. Nonlinear SINR and the corresponding link rates are evaluated for each finite power/template profile. Conditional on these choices, airtime variables $a_{uc} \geq 0$ minimise normalised transmit energy $\sum_{u,c} p_c a_{uc}$ subject to $\sum_u a_{uc} \leq 1$ per cell and $\sum_c R_{uc} a_{uc} \geq d_u$ per user; non-associated entries are fixed to zero. Each conditional problem is a linear program. The minimum over all feasible profiles is therefore the global optimum of this restricted finite-profile problem, up to solver tolerance. Each of six instances has 256 finite profiles. These are not S_{IF} gaps or full-plant bounds.

Allowing each user to split the delivered rate between cells enlarges the feasible set. Minimising that linear program (LP) over the same finite power/template profiles gives a lower bound on the restricted single-association optimum. The gap is the integer feasible optimum minus that lower bound. Seeds 6000000 through 6000005 generate the six declared small instances; channel gains, demands, the optimal profile, airtimes and residuals are supplied. All six have zero gap at the displayed precision. This is an observed property of these low-load restricted instances, not evidence that the complete stochastic plant is convex. The benchmark omits mobility, queues, uncertainty and continuous power optimisation and does not bound the learned-surrogate controller's performance gap. A certified optimum or relaxation-error bound for the complete plant remains unestablished.

Table 33. Certified restricted benchmark: fractional-association LP lower bound and exhaustive single-association optimum for normalised transmit energy.

Seed	Profiles	Feasible Profiles	LP Lower Bound	Integer Optimum	Gap
6000000	256	182	0.48	0.48	**0.00**
6000001	256	156	0.30	0.30	**0.00**
6000002	256	154	0.45	0.45	**0.00**
6000003	256	218	0.31	0.31	**0.00**
6000004	256	202	0.44	0.44	**0.00**
6000005	256	200	0.38	0.38	**0.00**

Notes: Bold values indicate zero gaps at the reporting precision in each independently seeded problem, which are the best displayed gaps. Objective values and profile counts from different problems are not ranked against one another.

6.5.2. Coordination-Layer Benchmark with Non-Shareable Local Models

The plant campaign cannot show what the shared-space maps G_i contribute, because its three resource agents belong to one administrative domain, every comparator has access to the complete telemetry and the trust gate acts on advisory peers and not on the coordination messages. A separate coordination-layer benchmark is therefore reported, in which the operating problem of Equations (24)–(26) is solved for $N \in \{4, 8, 16, 32, 64\}$ administrative domains whose local models cannot be shared; its configuration is listed in Table 11. Each domain i holds a private objective of the form in Equation (27) with a local dimension d_i drawn uniformly from 2 to 6, a private box $X_i = [0.02, 1.00]^{d_i}$ and a map G_i that loads each local variable onto one of $p = 3$ shared pools with a coefficient in $[0.50, 1.50]$ and, with probability 0.30, onto a second pool with a coefficient in $[0.10, 0.50]$. The inverse-share weights are $a_{ij} = 0.10\, s_i u_{ij}$, with a log-normal domain scale s_i of log-standard deviation 0.75 and u_{ij} uniform on $[0.50, 1.50]$, the linear coefficients c_{ij} are uniform on $[0.20, 2.00]$ and $\gamma_{ij} = 0.05$. An intent priority $w_i \in \{1, 2, 4\}$ multiplies the whole local objective, so that it changes the marginal value of a resource under scarcity without changing the declared demand. The budget is $b = \beta \sum_i G_i x_i^{\text{free}}$ with $\beta = 0.60$, where x_i^{free} is the uncoupled local minimiser, so the pools are oversubscribed by the free demand. These settings were fixed on separate development seeds and the reported instances use seeds that are disjoint from every seed range of the plant campaigns. The benchmark is a convex resource-allocation study without a radio, mobility or traffic model and its cost gaps are not S_{IF} differences.

Four allocators are compared under the information constraint that a domain may export p-dimensional quantities only. The shared-space coordinator runs the PDHG updates of Equations (28)–(30) with $\tau = \sigma = 0.95/(\sum_i \|G_i\|_2^2)^{1/2}$, which satisfies the step-size condition of Proposition 2 from one published scalar per domain, together with the tolerance 10^{-4} of the plant campaign and an iteration limit of 20,000; only $G_i \bar{x}_i$ and λ cross a domain boundary. The coordinator and the reference optimum of the benchmark were cross-checked against the implemented coordinator and central solver of the package on nine shared instances, with relative objective discrepancies below 10^{-6}. The aggregate-only allocator receives the declared free demand $G_i x_i^{\text{free}} \in \mathbb{R}^p$ of every domain once, splits b in proportion to it and lets each domain solve its local problem inside the delegated share, which is the demand-proportional principle of the refinement-based comparators in Table 10. The equal split delegates b uniformly among the domains that load a pool. The central full-information solver of the same problem supplies the reference optimum and is not admissible under the constraint, because it requires the coefficients, the reference allocation, the bounds and the coupling column of every local variable, that is $(6 + p)d_i$ private scalars per domain. The cost gap is the relative excess of $\sum_i f_i(x_i)$ over the reference optimum in per cent, the coupling excess is the largest value of $\max_r [\sum_i G_i x_i - b]_r^+ / b_r$ over the seeds in per cent, the payload is modelled at eight bytes per float for one upload and

one download per iteration and the last column counts the private scalars that a domain must disclose to the central solver.

The means over twenty seeds are reported in Table 34. The shared-space coordinator matches the full-information objective to the reported numerical precision at every scale, with a cost gap of 0.00% and a coupling excess of at most 0.16% that is set by the stopping tolerance, without disclosure of local model parameters, local feasible-set parameters or raw telemetry. The aggregate-only allocator loses 2.27% at four domains and between 2.71% and 3.83% at 8 to 64 domains, whereas the equal split loses between 25.21% and 37.67%. The paired difference between the aggregate-only and shared-space gaps is positive in all twenty seeds of every configuration (exact sign-flip test with Holm correction over the seven configurations, $p < 0.01$). The advantage of shared-space over aggregate-only allocation grows with scarcity: at sixteen domains the aggregate-only loss is 0.83% for $\beta = 0.80$ and 7.42% for $\beta = 0.40$. A variant of the proportional rule that also receives the priority w_i of every domain is worse than the plain rule, with losses between 7.89% and 11.82% at $\beta = 0.60$, so the plain rule is retained as the stronger admissible comparator. The price of coordination is iterative signalling, namely 563.55 iterations and 26.42 KiB per domain at four domains, falling to 84.15 iterations and 3.94 KiB at 64 domains, against a single exchange of six floats for the aggregate-only allocator. The small loss of the demand-proportional rule is also consistent with the absence of a separation between the proposed controller and the PACIFISTA-style comparator, which uses the same principle, on the single-domain plant.

Table 34. Coordination-layer benchmark with non-shareable local models: cost gap to the full-information optimum over twenty seeds.

β	N	Shared-Space Gap (%)	Aggregate-Only Gap (%)	Equal-Split Gap (%)	Coupling Excess (%)	PDHG Iterations	KiB per Domain	Private Scalars (Central)
0.60	4	**0.00**	2.27	25.21	0.16	563.55	26.42	37.69
0.60	8	**0.00**	2.71	35.23	0.01	310.20	14.54	37.01
0.60	16	**0.00**	3.83	30.32	0.00	156.30	7.33	36.93
0.60	32	**0.00**	3.59	37.67	0.00	113.50	5.32	35.85
0.60	64	**0.00**	3.50	37.24	0.00	84.15	3.94	36.15
0.80	16	**0.00**	0.83	21.71	0.00	72.05	3.38	35.55
0.40	16	**0.00**	7.42	42.85	0.01	695.35	32.59	35.94

Notes: Bold values indicate the lowest displayed cost gap among the three admissible allocators in each row; rows are separate comparison groups. Coupling excess, iterations and payload refer to the shared-space coordinator and the last column to the central reference; these diagnostics are not ranked.

The second part of the benchmark places the trust gate on the coordination channel, as specified in Algorithm 2 and not on advisory peers. Sixteen domains are coordinated for 40 epochs; the logarithm of every a_{ij} follows a first-order autoregressive process with coefficient 0.80 and innovation standard deviation 0.10 and the preceding realised allocation is used as x_i^{ref}. From epoch 10 onwards, a fraction of correctly authenticated domains seize the upper bounds of their local sets and report the contribution $\kappa G_i x_i$ with a falsification factor $\kappa < 1$, so that the common price is computed from an understated aggregate. The pools are physical: when the metered usage of a pool exceeds its budget, every variable that loads that pool is throttled in proportion. The trust model of Section 5.3 is used with its campaign parameters ($\alpha_w = 0.70$, initial score 0.80, threshold 0.60 and evidence weights 0.20, 0.40, 0.15 and 0.25 in Equation (43)), but its evidence is computed from behaviour and is not drawn from label-dependent distributions. Authentication quality is 1 because the adversary is an insider, consistency is one minus the relative ℓ_1 deviation between the reported contribution and the usage of the domain that the pool owner meters with 5% multiplicative noise, delivery is 1 when that deviation is within 10% and responsibility is

the share of a pool excess above 1% of capacity that is caused by consumption beyond the report. A rejected domain is capped at its delegated floor at the pools, that floor is reserved from b and the domain remains excluded until the end of the run. The reported quantities are the inflation of the honest-domain cost relative to the attack-free run of the same seed, evaluated over the same honest set and epochs 10 to 39, the mean and peak pool excess, the number of compromised domains detected over the twenty seeds, the delay from onset to rejection and the false rejections of honest domains.

The gate is switched on and off on identical seeds in Table 35. With 12.50% of the domains compromised and $\kappa = 0.20$, removal of the gate inflates the honest-domain cost by 24.48% and leaves a mean pool excess of 67.00%, whereas with the gate the inflation is 0.01% and the mean excess is 7.90%; 40 of 40 compromised domains are rejected after 3.52 epochs on average. With 25.00% compromised, the inflation is 50.95% without the gate and 2.26% with it (80 of 80 detected, mean delay 4.34 epochs). For $\kappa = 0.50$ the corresponding values are 36.11% and 3.85%, with a longer mean delay of 5.47 epochs. The paired difference between the two configurations is positive in all twenty seeds of these three cases (exact sign-flip test with Holm correction over the four cases, $p < 0.01$). The peak excess is equal with and without the gate because it is reached before the first rejection. No honest domain is rejected in any run, including the attack-free runs, in which the two configurations coincide. The gate's operating boundary is identified: under mild falsification ($\kappa = 0.80$) the evidence score of a compromised domain settles above the threshold, 3 of 40 compromised domains are rejected and containment is provided by the delegated bounds (inflation 171.70% without the gate and 110.50% with it, a difference that is positive in 2 of twenty seeds and not significant, $p = 0.50$); this boundary is the trade-off between threshold and evidence that is quantified in Table 26. Most of the seized resource is declared in this case and the declared seizure exhausts the budget in 90 and 48 of the 600 post-onset epochs of the two configurations, in which the Phase-I check fails and the honest domains are held at their floors. This damage is caused by the width of the delegated local sets and has to be contained by the delegated bounds of the refinement stage and not by the trust gate.

The benchmark isolates the coordination layer and its conclusions are limited accordingly. The local models are synthetic members of the family in Equation (27), the aggregate is recovered exactly, per-domain metering of pool usage is assumed to be available to the pool owner and the adversary follows one declared strategy. A plant-level experiment with many administrative domains, in which these effects are expressed in S_{IF}, remains future work.

Table 35. Contribution falsification on the coordination channel with the trust gate on and off (sixteen domains, twenty seeds).

Compromised	κ	Gate	Honest-Cost Inflation (%)	Mean Pool Excess (%)	Peak Pool Excess (%)	Detected	Delay (Epochs)	False Rejections	p Holm
12.50%	0.20	on	**0.01**	**7.90**	67.00	40/40	3.52	0	<0.01
12.50%	0.20	off	24.48	67.00	67.00	n/a	n/a	n/a	
25.00%	0.20	on	**2.26**	**17.56**	121.95	80/80	4.34	0	<0.01
25.00%	0.20	off	50.95	121.95	121.95	n/a	n/a	n/a	
12.50%	0.50	on	**3.85**	**7.64**	41.88	40/40	5.47	0	<0.01
12.50%	0.50	off	36.11	41.88	41.88	n/a	n/a	n/a	
12.50%	0.80	on	**110.50**	**16.55**	17.35	3/40	14.75	0	0.50
12.50%	0.80	off	171.70	17.35	17.35	n/a	n/a	n/a	

Notes: Within each attack case, bold values indicate the lower honest-cost inflation and the lower mean pool excess, including ties. Detection counts, delays, false rejections and p-values are not ranked; n/a denotes a quantity that does not exist when the gate is off or when no domain is detected. The Holm-adjusted p-value refers to the paired gate-off minus gate-on difference in inflation.

6.6. Scope, Limitations and Threats to Validity

The two evidence layers remain distinct: the archived legacy bridge records have a known configuration and three replications, but their flow fields reproduce the deterministic dry-run generator. They do not establish packet-level execution. The analytical study has complete seed and source manifests and remains a simulation on the development plant. The claims are restricted to these identified evidence sets.

The controllers were designed against the analytical plant used for evaluation. The original base campaign used seeds disjoint from its development seeds. However, those evaluation blocks had already been inspected before the later design-variant study. The base stack was evaluated on untouched seeds and the confirmatory reference is provided by it; a non-significant refinement is added on top of it by the selected variant, whose evaluation blocks had been inspected. Selection on the plant model also remains possible because a controller whose response model matches the plant abstraction may be advantaged. The comparator tuning reduces but does not eliminate this asymmetry, because the comparators' parameterisations were also designed by the authors. A further limitation is the reliance on a synthetic analytical plant; all generated per-epoch logs, configurations and scripts are part of the source package, which is available from the authors upon request. Future work will validate the framework on trace-driven public data such as the site-specific channels of DeepMIMO [68], the Madrid LTE traffic traces [69] and the O-RAN closed-loop resources of OpenRAN Gym [71], so that real channel and mobility dynamics enter the evaluation.

With respect to scale, the evaluation uses four cells and eighty users. The blueprint of Section 6.2 calls for three scales up to metro and multi-domain and the central architectural argument for distributed coordination is that its cost and its robustness scale with the number of coordinating agents. The package coordinates one agent per slice, so scaling the plant alone does not exercise the argument (Table 32) and scaling the number of intents up to twelve agents exercises it only in the regime where the central problem is still small, in which the distributed formulation incurs a higher cost than the centralised one (Table 31). The many-domain regime is therefore examined only at the coordination layer (Section 6.5.2), where the shared-space coordinator matches the full-information objective to the reported numerical precision for 4 to 64 domains without disclosure of local models, at the cost of iterative signalling. That benchmark has no radio, mobility or traffic model, its local models are synthetic and its trust evidence presupposes per-domain metering of pool usage, so a plant-level many-domain validation expressed in S_{IF} remains future work.

Diagnostic what-if calls also advance the twin random stream and update its stored forecast in the retained implementation. Counterfactual inspection therefore needs to be separated from state-changing prediction before a future rerun is used for isolated forecast or solver attribution. The present comparisons describe the retained controller stacks.

For mechanism attribution, primary ablation conclusions follow the corrected seed-level tests rather than the sign of a small uncorrected difference. By null effects in this finite campaign, it is shown that the safeguards are score-neutral on this plant, not that they are irrelevant; their effect is demonstrated where they are load-bearing (Section 6.5.2). Synthetic evidence separability, fixed resource floors, module interactions and limited overload duration can all affect those estimates. Capped-state boundedness, finite attack counts and a resource certificate are different properties and none substitutes for a stability or security proof.

Regarding the statistics, twenty independent seed blocks improve precision but do not imply a universal minimum detectable effect. Power depends on paired variance and multiplicity. The primary family contains seventeen pooled contrasts. Per-scenario and stress-scenario tests, the mixed model and weight rescoring are additional analyses with their stated correction scope and are interpreted as exploratory. A nonsignificant difference is not an equivalence result. The 60-epoch horizon limits claims about persistent overload, recovery and long-run constraints.

7. Discussion

In this section, the results are interpreted, and the evaluation is connected back to the architecture and formal assumptions. The layered design and its interpretability are discussed first, followed by the trade-off between centralisation and distribution in the light of the coordination-layer benchmark, the alignment of the framework with the 3GPP, ETSI and O-RAN specifications, the scope within which the formal guarantees apply and the practical questions that remain open for deployment.

7.1. Layered Design and Interpretability

A key message of this work is that intent-driven mobile networking is neither purely semantic nor purely algorithmic. A typed intent interface without a realisation theory is too weak, and a distributed solver without intent semantics is too disconnected from operator goals. The four-layer architecture makes both sides explicit: the business layer expresses the operator outcome, the service layer performs intent refinement into service-level requirements, the domain layer coordinates coupled resource intents, and the execution layer handles fast-timescale local control and safety. Refinement is thus a function of the service layer rather than an additional fifth architectural layer. This decomposition is especially important in mobile networks because RAN, transport, MEC and core are tightly coupled but not naturally centralised.

The interpretability and adaptivity trade-off is also central to the proposed design. Agentic systems promise flexible reasoning, but telecom operations require predictability. The framework therefore allows free-form natural-language reasoning only before the typed control boundary. Every LLM-derived object is schema-validated, re-typed and checked against the agent capability envelope before it may enter optimisation or invoke a tool; the LLM cannot actuate the network directly. Actual actuation remains bounded by typed policies, convex budget updates, finite template games or local MPC-like safety filters. This is more conservative than some generic agentic-networking visions, but it is a better fit for operator-grade systems.

7.2. Centralisation Versus Distribution

It is not claimed that all intelligence should be distributed. Functions that require a global and consistent view, are updated slowly and carry accountability, namely governance, audit and long-horizon planning, are kept centralised or at least globally visible. Functions that are latency-sensitive, depend on local state or touch privacy-sensitive data, namely state assimilation, fast actuation, fine-grained adaptation and local analytics, are distributed. This boundary is encoded in the mathematics: the slow-timescale primal-dual program represents the globally coupled decisions that benefit from coordination and the fast-timescale Lyapunov and MPC execution represents the local decisions that must be taken without waiting for a global round. The central full-information solver of Section 6 is the operating point at which the coupled decisions are also centralised; the campaign compares the measured costs and outcomes of the two complete controller stacks on a single-domain plant, where a full-information solver is admissible, and the distributed

formulation is therefore more costly without being better. The case for distribution rests on the regime in which that solver is not admissible. In the coordination-layer benchmark of Section 6.5.2, the shared-space coordinator reproduces the full-information optimum from p-dimensional exchanges alone, the aggregate-only alternative loses 2.27% to 3.83% and falsified contributions inflate the honest-domain cost by 24.48% to 50.95% unless trust admission is active; Section 6.6 records that these effects are not yet expressed at plant level.

7.3. Standards Alignment

The framework is expressed in terms that standardised systems already represent: 3GPP intent objects and analytics functions anchor typed intent handling and observability, ETSI ZSM and ENI supply the architectural language for closed loops, digital twins, translation, conflict detection and agents and O-RAN provides RIC-layer actuation points and public service models. The alignment is conceptual and functional, so that existing interfaces, information models and management services can be reused rather than a parallel management architecture being required; the evaluated prototype does not demonstrate conformance with any cited interface, as Table 2 makes explicit and practical adoption would still require implementation, integration, conformance testing and deployment-specific validation.

7.4. What the Theory Does and Does Not Guarantee

The theoretical guarantees are conditional statements. Hierarchical preservation requires sound semantic refinement and joint child feasibility. The stated primal-dual convergence result requires a saddle point, an exact aggregate and the declared step-size condition. An inexact aggregate requires a separate error analysis. The drift-plus-penalty result requires uncapped queues, bounded increments, a feasible stationary comparison policy and exact or bounded-error slotwise minimisation. Potential-game convergence requires finite actions and symmetric penalties. These conditions must be checked individually. The finite composition examples, resource certificates, regression traces and measured outcomes establish different parts of the implementation contract. Clipped controller deficits, learned forecasts and a short simulation do not establish the remaining assumptions.

The design-contract reading deserves to be made precise, because it determines how the guarantees should be used in practice. Each proved property is paired with explicit assumptions and applies only when those assumptions and the stated algorithmic conditions are satisfied. Hierarchical fulfilment preservation is thus a contract on refinement soundness together with joint child feasibility, distributed convergence is a contract on the saddle-point and PDHG step-size conditions, the Lyapunov bound is a contract on bounded increments, a feasible stationary comparison policy and exact or explicitly bounded-error slotwise drift-plus-penalty minimisation and potential-game convergence is a contract on finite action sets and symmetric conflict penalties. Approximate satisfaction of an assumption does not automatically yield an approximate guarantee; the resulting robustness must instead be established analytically or evaluated empirically. When the assumptions are materially violated, the corresponding proposition should not be treated as applicable. Presenting the assumptions explicitly allows an engineer to determine which formal results are applicable to a given deployment and which properties must instead be assessed or monitored empirically.

7.5. Practical Open Questions

Several practical questions remain open and should be documented explicitly: how rich the intent language should be before ambiguity becomes operationally harmful; how much of the digital-twin prediction stack should be model-based versus learned; how strict trust gating should be in multi-vendor deployments; how aggressively local ex-

ecutors should favour stability over raw utility; and which domains are best handled by convex optimisation, which by template games and which by local MPC. These are deployment-dependent model-selection questions whose appropriate answers cannot be fixed independently of the operational context.

Each of these questions is a model-selection decision whose best answer depends on the deployment context rather than on the framework itself. The richness of the intent language trades expressive power against the risk of ambiguity, the balance between model-based and learned prediction trades interpretability against adaptivity, the strictness of trust gating trades robustness against the risk of excluding honest peers, the aggressiveness of local executors trades raw utility against stability and the choice between convex optimisation, template games and local MPC trades modelling effort against control quality. The framework is designed to accommodate a range of deployment-specific choices, while the evaluation blueprint provides a structured means of comparing those choices using operational evidence. The open questions therefore define a structured agenda for deployment-specific evaluation and future research.

8. Conclusions and Future Work

Shared-space coordination has been developed as the central mechanism of a formal, standards-aligned framework for distributed artificial intelligence in agentic intent-based networking for mobile networks: domain agents with heterogeneous local decisions are coordinated through maps G_i into a common resource space, after a separate feasibility certificate and behind a trust-admission gate, so that local models and telemetry remain inside their domains. The core design choice is to model mobile-network autonomy as a hierarchy of bounded agents that refine, coordinate and execute intents across business, service, domain and execution layers, informed by relevant concepts and mechanisms in 3GPP, ETSI ZSM/ENI and O-RAN specifications and studies.

Formal definitions for agents, intents, intent hierarchies, network state and mobility state have been introduced. Intent realisation has then been formulated as a constrained stochastic multi-agent optimisation problem and a practical stack of solution layers has been derived: sound hierarchical refinement, distributed primal-dual coordination, Lyapunov-inspired deficit feedback with explicit hard resource checks, finite potential-game conflict resolution and an optional MPC-inspired execution layer. Under explicit assumptions, known results are instantiated for hierarchical fulfilment preservation under joint feasibility, primal-dual coordination convergence, Lyapunov stability and near-optimality for exact or bounded-error slotwise drift-plus-penalty control and potential-game convergence for discrete conflict mediation. The Lyapunov $O(1/V)$ bound is inherited by the optional H-step receding-horizon executor whenever the stated per-slot condition is satisfied by its applied first action; otherwise the executor is operated as a safety-filtered heuristic. Three algorithms have been proposed for hierarchical intent refinement, trust-aware distributed coordination and mobility-aware safe execution, together with an indicative analysis of their computational scaling. Security, privacy and trust have been integrated into the architecture rather than treated as afterthoughts. Finally, a reproducible evaluation blueprint has been defined and recovered bridge dry-run records have been reported for an ambiguous disaster-recovery intent scenario. Their configurations and replication metadata are retained. Trust rejections and safety interventions are observed in the legacy Python workflow, whose archived pipeline ranks last on the bridge endpoint, whereas the current stack executed through the same bridge workflow ranks first; the flow records of both reproduce a deterministic generator and are not presented as packet-level validation. The archived conflict counters of the legacy records are zero because the legacy detector used

the 1.0 threshold identified in Section 6.5; the counters are reported by the updated detector in Table 24.

The evidence for the mechanism is a coordination-layer benchmark with 4 to 64 administrative domains and twenty seeds. The shared-space coordinator matches the full-information objective to the reported numerical precision (cost gap 0.00%) without disclosure of local models, whereas, at the principal $\beta = 0.60$ setting, the best aggregate-only allocation loses 2.27% to 3.83% and an equal split loses 25.21% to 37.67%, at the price of 84.15 to 563.55 iterations of p-dimensional signalling per epoch. When 12.50% to 25.00% of the domains understate their contributions fivefold, removal of the trust gate inflates the honest-domain cost by 24.48% to 50.95%, against 0.01% to 2.26% with the gate; mild falsification ($\kappa = 0.80$) is not reliably detected by the gate, so that its containment depends on the delegated resource bounds. In addition, an analytical campaign has been evaluated on twenty held-out seed blocks after nine-point development grids for the base proposed controller and two tunable comparators. A backlog-aware radio floor with a latency-critical margin uplift was selected in a later design-variant study. That study is reported as exploratory because its evaluation blocks had already been inspected in the base campaign; the base design is thereby confirmed to lie near the plateau of this plant. Both tuned and untuned configurations of PACIFISTA-style mitigation and tabular multi-agent reinforcement learning are included. The selected DAI-BDIX and tuned PACIFISTA-style comparator both attain a pooled mean score of 0.80 at two-decimal precision; their paired difference is not significant after Holm correction over the full seventeen-contrast family. Significant positive differences are obtained against seven comparator configurations, including both MARL configurations. The differences against untuned PACIFISTA-style, the central full-information solver and the Confucius-style workflow are positive but not significant after correction. On this plant the controller is therefore significantly higher than seven of the eleven comparator configurations on score and not significantly separated from the remaining four after multiplicity correction; it is best on mean latency and the only configuration in which trust admission, certified feasibility and bounded controller state are combined; the tie with the tuned PACIFISTA-style comparator is not read as equivalence and the benefit of distributing the solver is established at the coordination layer, where the full-information objective is matched to the reported numerical precision by no other admissible allocator. The mechanism traces expose resource-certification status, predictive KPI warnings, actual conflict handling and the limits of synthetic trust evidence. A bounded controller state under sustained overload is guaranteed by the clipped controller deficits; physical queue stability is a plant property for which packet-level measurement is required. The implementation is supported as an analytical prototype that has been examined in depth, with measured costs and explicitly stated limitations. Further evaluation requires packet-level evidence tied to the updated controller and its simulator provenance, an independent plant, otherwise matched solver comparisons, plant-level many-domain coordination and a defensible bound on the complete stochastic realisation problem. The legacy records remain evidence of workflow execution.

Author Contributions: Conceptualization, I.I. and V.V.; methodology, I.I., S.N.-J. and S.F.; software, I.I. and V.B.; validation, I.I., V.B. and M.G.; formal analysis, I.I. and M.M.; investigation, I.I., S.F. and M.G.; resources, S.N.-J. and V.V.; data curation, I.I. and V.B.; writing (original draft preparation), I.I.; writing (review and editing), S.N.-J., S.F., V.B., M.G., M.M. and V.V.; visualization, I.I. and M.G.; supervision, V.V. and M.M.; project administration, I.I. and S.N.-J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data and source code supporting this study are available upon request to the authors. The package includes the analytical simulation code, the coordination-layer benchmark script with its records, effective configurations, seed and software manifests, raw analytical logs and scripts used to regenerate the current analytical tables and figures. The archived legacy bridge configuration, per-run and per-flow records, bridge source and dry-run provenance check are also included. These recovered records are not represented as verified packet-level measurements.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. 3rd Generation Partnership Project. *Architecture Enhancements for 5G System (5GS) to Support Network Data Analytics Services*; Technical Specification 3GPP TS 23.288 V18.6.0; ETSI/3GPP: Sophia Antipolis, France, 2024.
2. O-RAN ALLIANCE WG1. *O-RAN Empowering Vertical Industry: Scenarios, Solutions and Best Practice*; Technical Report; O-RAN ALLIANCE: Alfter, Germany, 2023.
3. Habibi, M.A.; Han, B.; Saimler, M.; Pavon, I.L.; Schotten, H.D. Towards an AI/ML-Driven SMO Framework in O-RAN: Scenarios, Solutions, and Challenges. In *2024 IEEE Future Networks World Forum (FNWF)*; IEEE: Piscataway, NJ, USA, 2024; pp. 7–14. [CrossRef]
4. Clemm, A.; Ciavaglia, L.; Granville, L.Z.; Tantsura, J. RFC 9315: Intent-Based Networking—Concepts and Definitions. RFC. IRTF Informational. 2022. Available online: https://www.rfc-editor.org/info/rfc9315/ (accessed on 27 September 2026).
5. 3rd Generation Partnership Project. *Management and Orchestration; Intent Driven Management Services for Mobile Networks*; Technical Specification 3GPP TS 28.312 V18.7.0; ETSI/3GPP: Sophia Antipolis, France, 2025.
6. Ioannou, I.; Vassiliou, V.; Christophorou, C.; Pitsillides, A. Distributed Artificial Intelligence Solution for D2D Communication in 5G Networks. *IEEE Syst. J.* **2020**, *14*, 4232–4241. [CrossRef]
7. Wang, J.; Zhang, L.; Yang, Y.; Zhuang, Z.; Qi, Q.; Sun, H.; Lu, L.; Feng, J.; Liao, J. Network Meets ChatGPT: Intent Autonomous Management, Control and Operation. *J. Commun. Inf. Netw.* **2023**, *8*, 239–255. [CrossRef]
8. Wang, Z.; Lin, S.; Yan, G.; Ghorbani, S.; Yu, M.; Zhou, J.; Hu, N.; Baruah, L.; Peters, S.; Kamath, S.; et al. Intent-Driven Network Management with Multi-Agent LLMs: The Confucius Framework. In *Proceedings of the ACM SIGCOMM Conference*; Association for Computing Machinery: New York, NY, USA, 2025; pp. 347–362. [CrossRef]
9. Chambolle, A.; Pock, T. A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. *J. Math. Imaging Vis.* **2011**, *40*, 120–145. [CrossRef]
10. Neely, M.J. *Stochastic Network Optimization with Application to Communication and Queueing Systems*; Synthesis Lectures on Communication Networks, Morgan & Claypool; Springer: Cham, Switzerland, 2010. [CrossRef]
11. Monderer, D.; Shapley, L.S. Potential Games. *Games Econ. Behav.* **1996**, *14*, 124–143. [CrossRef]
12. Li, C.; Havel, O.; Olariu, A.; Martinez-Julia, P.; Nobre, J.C.; Lopez, D.R. RFC 9316: Intent Classification. RFC. IRTF Informational. 2022. Available online: https://www.rfc-editor.org/info/rfc9316/ (accessed on 27 September 2026).
13. 3rd Generation Partnership Project. *Study on Enhanced Intent Driven Management Services for Mobile Networks*; Technical Report 3GPP TR 28.912 V18.0.1; ETSI/3GPP: Sophia Antipolis, France, 2024.
14. 3rd Generation Partnership Project. *Management and Orchestration; Management Data Analytics (MDA)*; Technical Specification 3GPP TS 28.104 V18.5.0; ETSI/3GPP: Sophia Antipolis, France, 2024.
15. 3rd Generation Partnership Project. *Management and Orchestration; Artificial Intelligence/Machine Learning (AI/ML) Management*; Technical Specification 3GPP TS 28.105 V18.5.0; ETSI/3GPP: Sophia Antipolis, France, 2024.
16. 3rd Generation Partnership Project. *Security Architecture and Procedures for 5G System*; Technical Specification 3GPP TS 33.501 V18.5.0; ETSI/3GPP: Sophia Antipolis, France, 2024.
17. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Reference Architecture*; Group Specification ETSI GS ZSM 002 V1.1.1; ETSI: Sophia Antipolis, France, 2019.
18. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Closed-Loop Automation; Part 1: Enablers*; Group Specification ETSI GS ZSM 009-1 V1.1.1; ETSI: Sophia Antipolis, France, 2021.
19. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Intent-Driven Autonomous Networks; Generic Aspects*; Group Report ETSI GR ZSM 011 V1.1.1; ETSI: Sophia Antipolis, France, 2023.
20. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Intent-Driven Closed Loops*; Group Specification ETSI GS ZSM 016 V1.1.1; ETSI: Sophia Antipolis, France, 2024.

21. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Network Digital Twin for Enhanced Zero-Touch Network and Service Management*; Group Specification ETSI GS ZSM 018 V1.1.1; ETSI: Sophia Antipolis, France, 2024.
22. ETSI ISG ZSM. *Zero-Touch Network and Service Management (ZSM); Study on the Utilization of Agents in Autonomous Networks*; Group Report ETSI GR ZSM 020 V1.1.1; ETSI: Sophia Antipolis, France, 2026.
23. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); System Architecture*; Group Specification ETSI GS ENI 005 V3.1.1; ETSI: Sophia Antipolis, France, 2023.
24. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); Intent Policy Model Gap Analysis*; Group Report ETSI GR ENI 013 V1.1.1; ETSI: Sophia Antipolis, France, 2023.
25. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); Definition, Requirements and Procedure of Intent Policy Multi-Stage Translating*; Group Specification ETSI GS ENI 033 V4.1.1; ETSI: Sophia Antipolis, France, 2024.
26. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); Requirements and Detailed Procedure of Network Policy Conflict Detection*; Group Specification ETSI GS ENI 034 V4.1.1; ETSI: Sophia Antipolis, France, 2024.
27. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); Study on AI Agents Based Next-Generation Network Slicing*; Group Report ETSI GR ENI 051 V4.1.1; ETSI: Sophia Antipolis, France, 2025.
28. ETSI ISG ENI. *Experiential Networked Intelligence (ENI); Study on Multi-Agent Frameworks for Next-Generation Core Networks*; Group Report ETSI GR ENI 056 V4.1.1; ETSI: Sophia Antipolis, France, 2025.
29. O-RAN ALLIANCE. *O-RAN Specifications*; O-RAN ALLIANCE: Alfter, Germany, 2026.
30. O-RAN ALLIANCE. *67 New or Updated O-RAN Technical Documents Released Since November 2024*; O-RAN ALLIANCE: Alfter, Germany, 2025.
31. Autio, C.; Schwartz, R.; Dunietz, J.; Jain, S.; Stanley, M.; Tabassi, E.; Hall, P.; Roberts, K. *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile*; Technical Report NIST AI 600-1; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2024. [CrossRef]
32. OWASP GenAI Security Project. *LLM01:2025 Prompt Injection*; OWASP GenAI Security Project: Wakefield, MA, USA, 2025.
33. Leivadeas, A.; Falkner, M. A Survey on Intent-Based Networking. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 625–655. [CrossRef]
34. Velasco, L.; Signorelli, M.; de Dios, O.G.; Papagianni, C.; Bifulco, R.; Olmos, J.J.V.; Pryor, S.; Carrozzo, G.; Schulz-Zander, J.; Bennis, M.; et al. End-to-End Intent-Based Networking. *IEEE Commun. Mag.* **2021**, *59*, 106–112. [CrossRef]
35. Gomes, P.H.; Buhrgard, M.; Harmatos, J.; Mohalik, S.K.; Roeland, D.; Niemöller, J. Intent-driven Closed Loops for Autonomous Networks. *J. ICT Stand.* **2021**, *9*, 257–290. [CrossRef]
36. Gritli, N.; Khendek, F.; Toeroe, M. Decomposition and Propagation of Intents for Network Slice Design. In *IEEE 4th 5G World Forum*; IEEE: Piscataway, NJ, USA, 2021; pp. 165–170. [CrossRef]
37. Xie, M.; Gomes, P.H.; Niemöller, J.; Waldemar, J.P. Intent-Driven Management for Multi-Vertical End-to-End Network Slicing Services. In *IEEE Globecom Workshops*; IEEE: Piscataway, NJ, USA, 2022; pp. 1285–1291. [CrossRef]
38. Banerjee, A.; Mwanje, S.S.; Carle, G. An Intent-Driven Orchestration of Cognitive Autonomous Networks for RAN Management. In *2021 17th International Conference on Network and Service Management*; IEEE: Piscataway, NJ, USA, 2021; pp. 380–384. [CrossRef]
39. Costa, A.M.D.; Murillo, L.M.C. Integration of Network Slice Controller for Enhanced Intent-Based Networking in 5G/6G Networks. In *Proceedings of the 18th Workshop on Mobility in the Evolving Internet Architecture*; Association for Computing Machinery: New York, NY, USA, 2023; pp. 31–36. [CrossRef]
40. Sharma, Y.; Bhamare, D.; Sastry, N.; Javadi, B.; Buyya, R. SLA Management in Intent-Driven Service Management Systems: A Taxonomy and Future Directions. *ACM Comput. Surv.* **2023**, *55*, 292. [CrossRef]
41. Manias, D.M.; Chouman, A.; Shami, A. Towards Intent-Based Network Management: Large Language Models for Intent Extraction in 5G Core Networks. In *2024 20th International Conference on the Design of Reliable Communication Networks (DRCN)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [CrossRef]
42. Ayoub, O.; Natalino, C.; Al Housseini, A.; Foschum, F.; Morger, P.; Leidi, T.; Hock, D.; Monti, P. Explanation-Based Runtime Verification for Trustworthy ML-Driven Optical Networks. *arXiv* **2026**, arXiv:2607.20675. [CrossRef]
43. Sana, M.; Domenico, A.D.; Strinati, E.C.; Clémente, A. Multi-Agent Deep Reinforcement Learning for Distributed Handover Management in Dense mmWave Networks. In *ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing*; IEEE: Piscataway, NJ, USA, 2020; pp. 8976–8980. [CrossRef]
44. Sana, M.; Domenico, A.D.; Yu, W.; Lostanlen, Y.; Strinati, E.C. Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 6520–6534. [CrossRef]
45. Ioannou, I.; Christophorou, C.; Vassiliou, V. AI/ML for D2D Communication in 5G/6G Networks and Beyond: Concept, Applications, Challenges, and Future Directions. *TechRxiv* **2025**. [CrossRef] [PubMed]
46. Houidi, O.; Bakri, S.; Zeghlache, D. Multi-Agent Graph Convolutional Reinforcement Learning for Intelligent Load Balancing. In *NOMS 2022—2022 IEEE/IFIP Network Operations and Management Symposium*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6. [CrossRef]

47. Moayyedi, A.; Ahmadi, M.; Salahuddin, M.A.; Boutaba, R.; Saleh, A. Generalizable GNN-Based 5G RAN/MEC Slicing and Admission Control in Metropolitan Networks. In *NOMS 2023—2023 IEEE/IFIP Network Operations and Management Symposium*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–9. [CrossRef]
48. Zhou, G.; Zhao, L.; Zheng, G.; Xie, Z.; Song, S.; Chen, K.C. Joint Multi-Objective Optimization for Radio Access Network Slicing Using Multi-Agent Deep Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2023**, *72*, 11828–11843. [CrossRef]
49. Fawaz, H.; Houidi, O.; Zeghlache, D.; Lesca, J.; Quang, P.T.A.; Leguay, J.; Medagliani, P. Graph Convolutional Reinforcement Learning for Load Balancing and Smart Queuing. In *2023 IFIP Networking Conference (IFIP Networking)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–9. [CrossRef]
50. Chiputa, M.; Zhang, M.; Ali, G.G.M.N.; Chong, P.H.J.; Sabit, H.; Kumar, A.; Li, H. Enhancing Handover for 5G mmWave Mobile Networks Using Jump Markov Linear System and Deep Reinforcement Learning. *Sensors* **2022**, *22*, 746. [CrossRef] [PubMed]
51. del Prever, P.B.; D'Oro, S.; Bonati, L.; Polese, M.; Tsampazi, M.; Lehmann, H.; Melodia, T. PACIFISTA: Conflict Evaluation and Management in Open RAN. *IEEE Trans. Mob. Comput.* **2025**, *24*, 10590–10605. [CrossRef]
52. Zacarias, I.; Grunewald, M.; Gentzen, F.; Masip-Bruin, X.; Jukan, A. Enhancing Secure Intent-Based Networking with an Agentic AI: The EU Project MARE Approach. *arXiv* **2026**, arXiv:2604.06856. [CrossRef]
53. Singh, A.K.; Nguyen, K.K. Communication Efficient Compressed and Accelerated Federated Learning in Open RAN Intelligent Controllers. *IEEE/ACM Trans. Netw.* **2024**, *32*, 3361–3375. [CrossRef]
54. Javed, F.; Mangues-Bafalluy, J.; Zeydan, E.; Blanco, L. Trustworthy Reputation for Federated Learning in O-RAN Using Blockchain and Smart Contracts. *IEEE Open J. Commun. Soc.* **2025**, *6*, 1343–1362. [CrossRef]
55. Mayne, D.Q.; Rawlings, J.B.; Rao, C.V.; Scokaert, P.O.M. Constrained model predictive control: Stability and optimality. *Automatica* **2000**, *36*, 789–814. [CrossRef]
56. Mamdani, E.H.; Assilian, S. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *Int. J. Man-Mach. Stud.* **1975**, *7*, 1–13. [CrossRef]
57. National Institute of Standards and Technology. *Module-Lattice-Based Key-Encapsulation Mechanism Standard*; Federal Information Processing Standards Publications FIPS 203; NIST: Gaithersburg, MD, USA, 2024. [CrossRef]
58. National Institute of Standards and Technology. *Module-Lattice-Based Digital Signature Standard*; Technical Report FIPS 204; NIST: Gaithersburg, MD, USA, 2024. [CrossRef]
59. National Institute of Standards and Technology. *Stateless Hash-Based Digital Signature Standard*; Technical Report FIPS 205; NIST: Gaithersburg, MD, USA, 2024. [CrossRef]
60. ETSI. *Quantum Key Distribution (QKD); Protocol and Data Format of REST-Based Key Delivery API*; Group Specification GS QKD 014 V1.1.1; ETSI: Sophia Antipolis, France, 2019.
61. Kish, S.; Pieprzyk, J.; Camtepe, S. Trends in Quantum Key Distribution (QKD). In *Quantum Technologies*; Jang-Jaccard, J., Caroff, P., Blezinger, E., Mulder, V., Mermoud, A., Lenders, V., Eds.; Springer: Cham, Switzerland, 2026; pp. 119–131. [CrossRef]
62. Rescorla, E. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446. IETF Standards Track. 2018. Available online: https://www.rfc-editor.org/info/rfc8446/ (accessed on 27 September 2026).
63. CTTC. *5G-LENA*; CTTC: Castelldefels, Spain, 2026.
64. Lacava, A.; Bordin, M.; Polese, M.; Sivaraj, R.; Zugno, T.; Cuomo, F.; Melodia, T. ns-O-RAN: Simulating O-RAN 5G Systems in ns-3. In *Proceedings of the 2023 Workshop on ns-3 (WNS3)*; ACM: New York, NY, USA, 2023; pp. 35–44. [CrossRef]
65. Nardini, G.; Sabella, D.; Stea, G.; Thakkar, P.; Virdis, A. Simu5G—An OMNeT++ Library for End-to-End Performance Evaluation of 5G Networks. *IEEE Access* **2020**, *8*, 181176–181191. [CrossRef]
66. Polese, M.; Bonati, L.; D'Oro, S.; Johari, P.; Villa, D.; Velumani, S.; Gangula, R.; Tsampazi, M.; Robinson, C.P.; Gemmi, G.; et al. Colosseum: The Open RAN Digital Twin. *IEEE Open J. Commun. Soc.* **2024**, *5*, 5452–5466. [CrossRef]
67. srsRAN Project. *O-RAN NearRT-RIC and xApp Documentation*; srsRAN Project: Cork, Ireland, 2026.
68. Alkhateeb, A. DeepMIMO: A Generic Deep Learning Dataset for Millimeter Wave and Massive MIMO Applications. *arXiv* **2019**, arXiv:1902.06435. [CrossRef]
69. Pérez, P.F.; Fiandrino, C.; Widmer, J. Characterizing and Modeling Mobile Networks User Traffic at Millisecond Level. In *WiNTECH '23: Proceedings of the 17th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization* ; Association for Computing Machinery: New York, NY, USA, 2023; pp. 64–71. [CrossRef]
70. Andrade-Hoz, J.; Wang, Q.; Alcaraz-Calero, J.M. Infrastructure-Wide and Intent-Based Networking Dataset for 5G-and-Beyond AI-Driven Autonomous Networks. *Sensors* **2024**, *24*, 783. [CrossRef] [PubMed]
71. Bonati, L.; Polese, M.; D'Oro, S.; Basagni, S.; Melodia, T. OpenRAN Gym: AI/ML development, data collection, and testing for O-RAN on PAWR platforms. *Comput. Netw.* **2023**, *220*, 109502. [CrossRef]
72. Kraft, D. *A Software Package for Sequential Quadratic Programming*; Technical Report DFVLR-FB 88-28; DFVLR, Institut für Dynamik der Flugsysteme: Oberpfaffenhofen, Germany, 1988.

73. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. Correction in *Nat. Methods* **2020**, *17*, 352. [CrossRef] [PubMed]
74. Holm, S. A Simple Sequentially Rejective Multiple Test Procedure. *Scand. J. Stat.* **1979**, *6*, 65–70.
75. Benjamini, Y.; Hochberg, Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *J. R. Stat. Soc. Ser. B (Methodol.)* **1995**, *57*, 289–300. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.