Intelligent Data Analysis in Integrated Sensing And Communication Systems

Sima Barzegar[1*], Jaume Comellas[2], Luis Velasco[2], and Marc Ruiz[2]

(1) Barcelona Supercomputing Center (BSC), Barcelona, Spain

(2) Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

**e-mail: sima.barzegar@bsc.es*

ABSTRACT

The integration of optical communication and distributed sensing enables autonomous network operation, as well as increases business opportunities for network operators by opening the door to offer telemetry and sensing-as-a-service to their clients. In this paper, we introduce a distributed and hierarchical integrated telemetry and sensing architecture and explore two different use cases. First, we present a self-adaptive telemetry agent managing a dynamic ensemble of lightweight models via an online soft-weighting mechanism for robust modelling and forecasting. Additionally, a distributed agentic-based system combining feature extraction, artificial intelligence reasoning, and human knowledge achieves explainable state of polarization event classification.

Keywords: Integrated Sensing and Communications, Intelligent Network Operation

1. Introduction

The integration of optical communication and distributed sensing—collectively known as Integrated Sensing and Communications (ISAC)—is receiving significant global attention due to its ability to bring valuable additional functionalities to existing network infrastructures. By enabling advanced capabilities such as urban structure imaging, ocean seismic detection, infrastructure safety monitoring, and environmental event tracking, ISAC effectively paves the way for fully autonomous network operations. Furthermore, these capabilities open up new operational efficiencies and expanded business possibilities for network operators, allowing them to transform legacy infrastructures into high-precision monitoring environments through data streams like State-of-Polarization (SOP) measurements [1].

However, deploying effective ISAC frameworks introduces substantial technical complexity. These systems must seamlessly combine highly heterogeneous data sources—ranging from Distributed Acoustic Sensing (DAS) and SOP monitoring to packet-optical layer metrics—to achieve multifaceted visibility across both the packet and optical domains. Managing this diverse data requires a wide variety of processing steps distributed across a hierarchical architecture of local network sites and centralized hubs [2]. This workflow includes AI-driven automation for real-time feature extraction, data compression, and dimensionality reduction at the edge, alongside centralized human-in-the-loop supervision to calibrate context and knowledge rules for complex event reasoning. Balancing these automated and human-based processes is vital to address diverse, multi-domain use cases, including predictive failure management, Quality of Transmission (QoT) estimation, and native security protections against physical infrastructure threats and adversarial machine learning attacks [3].

To address these challenges, this paper explores intelligent and autonomous processing techniques for heterogeneous telemetry and environmental events collected from real data sources. On the one hand, a self-adaptive telemetry and sensing modelling agent is designed to provide robust, near-real-time time series modelling and forecasting under non-stationary environmental conditions. On the other hand, a distributed agentic-based SOP analysis system that integrates supervised feature extraction, centralized AI agent reasoning, and human operator knowledge is presented for SOP event classification for multiple operational purposes.

2. Reference Architecture

Figure 1 shows the reference integrated telemetry and sensing architecture, which is a hierarchical system designed to enhance the operation of end-to-end multilayer packet-optical networks by combining high-precision optical sensing with AI-driven automation. As depicted in the figure, the framework is structured across multiple distributed network sites that interact with a centralized management hub through a multi-tier data streaming layer.

At the local site level, the architecture incorporates diverse sensing technologies, which might include DAS and SOP monitoring, as well as other integrated optical sensing and communication modules. These devices ingest a wide variety of data samples to the local telemetry and sensing agent that performs key tasks such as initial data processing fueled with AI capabilities for feature extraction, dimensionality reduction, and data compression. This processed data

979-8-3195-4420-9/26/$31.00

feeds into local intelligence modules in charge of assisting near-real-time control decisions for local devices and systems.

Data from these distributed sites is aggregated at the centralized telemetry and sensing system which maintains a comprehensive overall network view. It stores both measurements and events generated and received from local agents, as well as by the centralized intelligence that processes inputs from different sites. By integrating data and knowledge across packet and optical domains, a multifaceted visibility into the network infrastructure and running services can be achieved. This global perspective enables advanced use cases such as end-to-end Quality of Service (QoS) scheduling, predictive failure management, and Quality of Transmission (QoT) estimation, to mention few.

Finally, native security is integrated at both the local and centralized levels. The system implements a defense-in-depth strategy to ensure the integrity and authenticity of telemetry and sensing data. This includes using optical sensing for security to detect physical infrastructure threats and deploying specific protections for AI models against adversarial attacks [3]. This coordinated architecture effectively provides "telemetry-as-a-service," allowing network operators to optimize performance and resilience across all network segments.

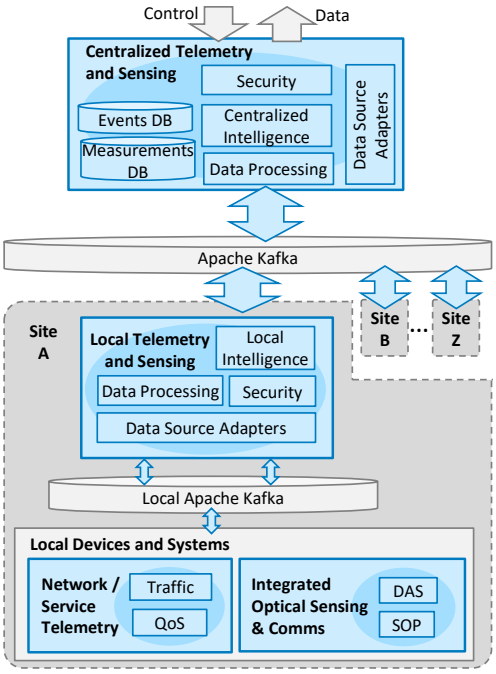

Figure 1 Integrated Telemetry and Sensing Architecture

3. Self-Adaptive Time Series Modelling Agent

In this section, we present a self-adaptive telemetry and sensing modelling agent that runs as part of the local intelligence component in the site-based local agent in Figure 1. This self-adaptive modelling agent is designed to provide robust, near-real-time time series modelling and forecasting. Instead of relying on a single model, which often degrades under the non-stationary conditions of real environments (e.g., regime shifts or variable noise), this agent manages a dynamic ensemble of lightweight base models in parallel. These models—including naive persistence, linear/polynomial regressions, and tracking filters like alpha-beta and Kalman [4]—are combined through a soft-weighting mechanism where their individual influence is adjusted online based on their recent performance. The agent consumes telemetry and/or sensing observations and produces enriched outputs that includes future-state predictions for a configurable time horizon. Moreover, by exposing internal interpretability signals such as the evolution of model weights and error rankings, the agent provides with not only a predicted value but also a measure of the current models' accuracy and adequacy. This enables the site-level management to make proactive, data-driven decisions while maintaining a fully auditable and reproducible operation loop.

Algorithm 1 presents the main procedure of the self-adaptive modelling agent for a generic telemetry or sensing data stream. Thus, let us assumed that this algorithm is invoked every time t a single measurement $y(t)$ is received. The procedure also receives the model ensemble Θ and the set of current weights W, and produces the prediction $\hat{y}(t+m)$ for the target time horizon $t+m$, as well as the updated W set.

The procedure consists in two main phases: *i*) weights update (lines 1-6 in Algorithm 1), and *ii*) forecasting (lines 7-10). In the first phase, the prediction done at time $t-m$ for current time t is retrieved and compared with the actual received value (line 1). Note that individual predictions for each of the models in the ensemble are stored, so model errors ε can be computed (lines 2-4). Then, the resultant error is used to update W, i.e., adding a weight reward that is inversely proportional to ε (line 5). Once the weights are updated, the prediction for $t+m$ is computed for each model (lines 6-8), and stored in the internal DB for a future weight update round (line 9). Finally, the algorithm returns as single prediction the weighted average of models' predictions according to W weights (line 10).

Algorithm 1 – Main Procedure

IN: $y(t)$, Θ, W **OUT:** $\hat{y}(t+m)$, W

1:	$\hat{y}(t) \leftarrow$ get(DB, "$\hat{y}$",t)
2:	$\varepsilon \leftarrow \emptyset$
3:	**for** $i == 0..\|\Theta\|$ **do**
4:	$\varepsilon_i \leftarrow$ compute_error($y(t)$, $\hat{y}_i(t)$)
5:	$W \leftarrow$ update_weights(W, ε)
6:	$\hat{y}(t+m) \leftarrow \emptyset$
7:	**for** $i == 0..\|\Theta\|$ **do**
8:	$\hat{y}_i(t+m) \leftarrow$ predict(Θ_i, $t+m$)
9:	set(DB, "$\hat{y}$",$t+m$, $\hat{y}(t+m)$)
10:	**return** weigthed_avg($\hat{y}(t+m)$, W), W

Figure 2 illustrates the performance of a Python-based implementation the self-adaptive modelling agent. As for the data stream to model, we selected traffic data series available in [5] as real network telemetry data. In particular, we selected traffic flow measurements of video streaming services, collected by a real network operator along one week. Using this data as baseline stream, we intentionally modified it to add increasing noise, thus moving from smooth to fluctuating, flickering time series data. Figure 2a shows the prediction returned by the ensemble Θ for time horizon m=1, whereas Figure 2b shows the evolution of weights for each of the 5 different models considered in this work. As can be observed, the proposed ensemble produces a robust, high accurate prediction regardless of the characteristics of the time series data. However, the evolution of the weights clearly reveals that the modelled data experience dynamic changes. This can be seen mostly after the dashed line, where the linear model is increasing (improving) its weight, reaching similar values than other models. Therefore, the accurate prediction is accompanied by an explanatory output revealing that the nature of the data is changing, which might trigger additional measures, e.g., to analyse whether the behavioural changes is due to a service anomaly, a failure, or an attack.

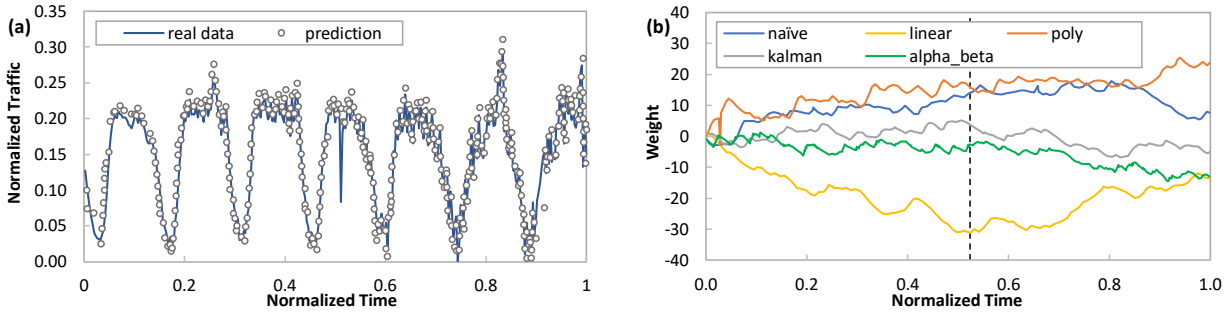

Figure 2 Adaptive Telemetry Agent Performance: prediction accuracy (a) and models' weight analysis (b)

4. Agentic-based SOP Analysis System

This section presents a distributed sensing architecture that combines supervised feature extraction, agentic-based sensing data analysis, and human supervision (Figure 3). The system collects SOP measurements from the receiver with a fine granular resolution e.g., at µs scale. The measures are received and processed at the local intelligence agent (labelled as 1 in Figure 3), which firstly buffers the sensing data stream. With a larger periodicity, e.g., every second, the feature extractor processes the data in the buffer, computing a number of relevant features that capture the behavior: *i*) mean, standard deviation, and correlation of the three Stoke parameters within the buffered time window; *ii*) the rotation speed in radians per second, and *iii*) the acceleration with respect to previous time window (2). Once the features are computed, they are sent to the centralized telemetry and sensing component, and the buffer is emptied.

The received features are then processed by a centralized component that executes an AI agent in charge of analyzing the input features and reasoning, with the help of the context and knowledge provided by the human operator, which event is observed. Without loss of generality, we consider three different events: *i*) *normal*, where SOP behaves as expected due to current conditions and context, *ii*) *outlier*, where the data behaves out of typical normality, although not experiencing a suspicious behavior; and *iii*) *anomaly*, where SOP features shows a clear unexpected behavior that might be consequence of a system failure, high environmental event, or attack. The estimation is produced with a metric that provides a level of likelihood of each case (3). The result of the reasoning

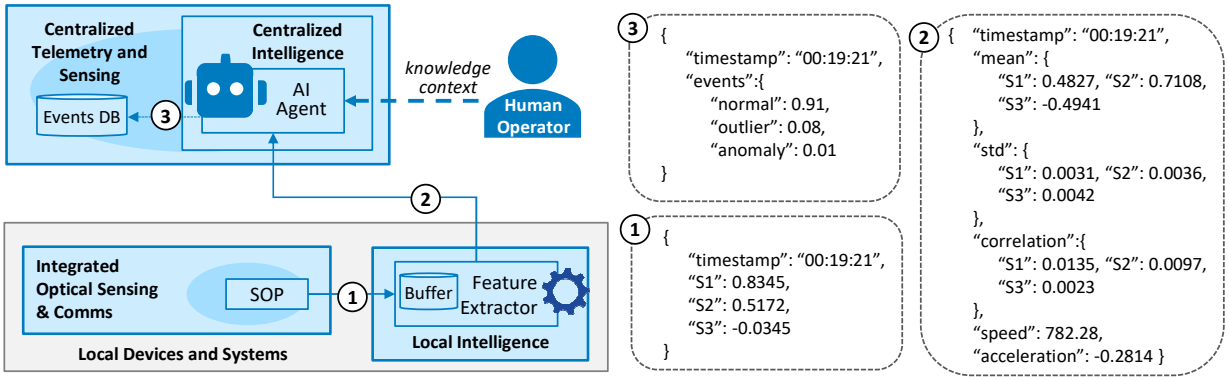

Figure 3 Distributed SOP analysis architecture

process is stored in an event database that can be consumed by other systems and actors such as executing agents for autonomous network operation and/or human operators for further analysis and calibration of context and knowledge.

In order to prove the feasibility of the proposed system, we implemented a proof-of-concept test consisting in a distributed system deployed in Docker (Figure 4). The distributed system consists of a first container that emulates the receiver generating the SOP measurements. To this aim, the experimental SOP data available in [6] have been used to emulate a realistic environment. A second container implements an influxDB buffer and a Python agent that performs feature extraction. Finally, a third container deploys the AI agent, that consists in a local instance of the Gemma2-9b model, from the Gemma 2 family built from the same research and technology used to create the Gemini models [7]. The agent has been instructed with knowledge based on a set of human-based rules needed to perform proper event classification.

The captures in Figure 4 have been obtained from the logging system of each of the components, and complement the message illustrated in Figure 3. Interestingly, we can see how the reasoning AI agent is capable to provide an event classification (in green), as well as an explanatory description (in blue) to understand which of the human-based rules provided as knowledge have been used to infer the returned event class. Therefore, explainability and traceability of agentic-based reasoning can be easily embedded as useful metadata of the event classification results, which clearly increases the added value of this AI-based approach of sensing data analysis.

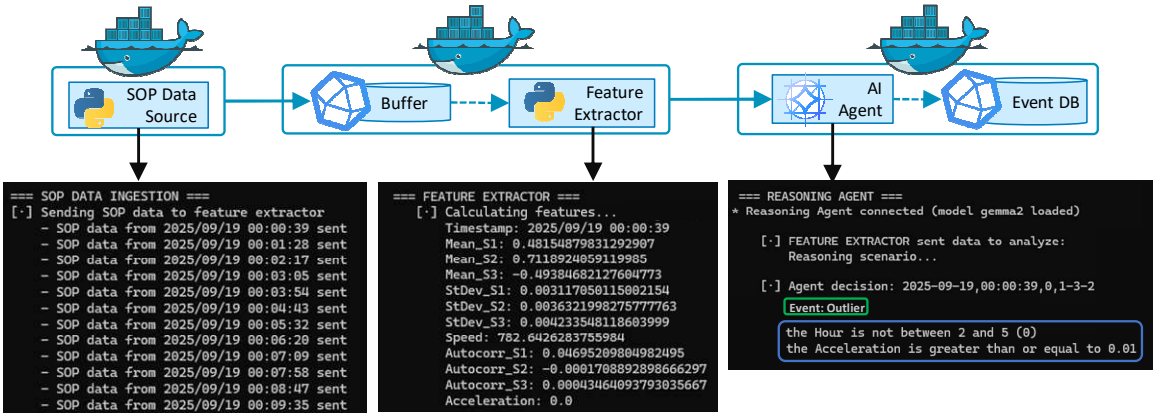

Figure 4 Proof-of-Concept Test

5. Conclusions

This work presented advanced autonomous processing techniques for ISAC environments, showcasing a self-adaptive telemetry agent for robust forecasting under non-stationary conditions and a distributed system for explainable SOP event classification based on AI agents. These innovations effectively pave the way for intelligent, fully autonomous network operations, as well as for expanded business possibilities for network operators.

ACKNOWLEDGEMENT

The authors want to thank the students Marc Guitart and Pau Quilez for their valuable contributions. The research leading to these results has received funding from the European Union's Horizon Europe research and innovation programme under G.A. No. 101092766 (ALLEGRO), the project PID2024-157824OB-I00 funded by MICIU/ AEI/10.13039/501100011033 /FEDER, UE and from ICREA.

REFERENCES

[1] J.C. Wang *et al.*, “Fiber Events Recognition in Integrated Sensing and Communication Systems Based on Signal State of Polarization Analysis,” in Proc. PIERS-Fall, 2025.

[2] L. Velasco *et al.*, "Distributed Intelligence for Pervasive Optical Network Telemetry," IEEE JOCN, vol. 15, 2023.

[3] M. Ruiz *et al.*, “Network digital twinning solutions to increase the security of multi-domain optical transport networks,” IEEE JOCN, vol. 18, 2026.

[4] J. -B. Kim *et al.*, “On the Design of Kalman Filter with Low Complexity for 6G-Based ISAC: Alpha and Alpha–Beta Filter Perspectives,” Electronics, vol. 14, 2025.

[5] M. Ruiz *et al.*, “Network traffic measurements for 4G, 5G and 6G services,” 2026. https://doi.org/10.34810/DATA2425

[6] S. Shen *et al.*, “Open-Source Data for Multi-Domain Network Monitoring and Sensing in Optical and Wireless Networks,” in Proc. OFC, 2025.

[7] Gemma 2 model card, https://huggingface.co/google/gemma-2-9b.