Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

June 2026 Volume XVIII Number 2 ISSN 2061-2079

Technically Co-Sponsored by

IEEE ComSoc IEEE Communications Society

hte

IEEE HUNGARY SECTION

Indexing information

Infocommunications Journal is covered by Inspec, Compendex and Scopus.
Infocommunications Journal is also included in the Thomson Reuters – Web of ScienceTM Core Collection, Emerging Sources Citation Index (ESCI)

Infocommunications Journal

Technically co-sponsored by IEEE Communications Society and IEEE Hungary Section

Supporters

FERENC BANCSICS – president, Scientific Association for Infocommunications (HTE)

MTA200

HTE Infocommunications Journal welcomes the 200th anniversary of the Hungarian Academy of Sciences and gratefully acknowledges its continuous support of the journal in recent years.

nmhh National Media and Infocommunications Authority | Hungary

The publication was also produced with the support of the NMHH.

Editorial Office (Subscription and Advertisements)**:**
Scientific Association for Infocommunications
H-1051 Budapest, Bajcsy-Zsilinszky str. 12, Room: 502
Phone: +36 1 353 1027 • E-mail: info@hte.hu • Web: www.hte.hu

Articles can be sent also to the following address:
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
Phone: +36 1 463 4189 • E-mail: pvarga@tmit.bme.hu

Subscription rates for foreign subscribers: 4 issues 13.700 HUF + postage

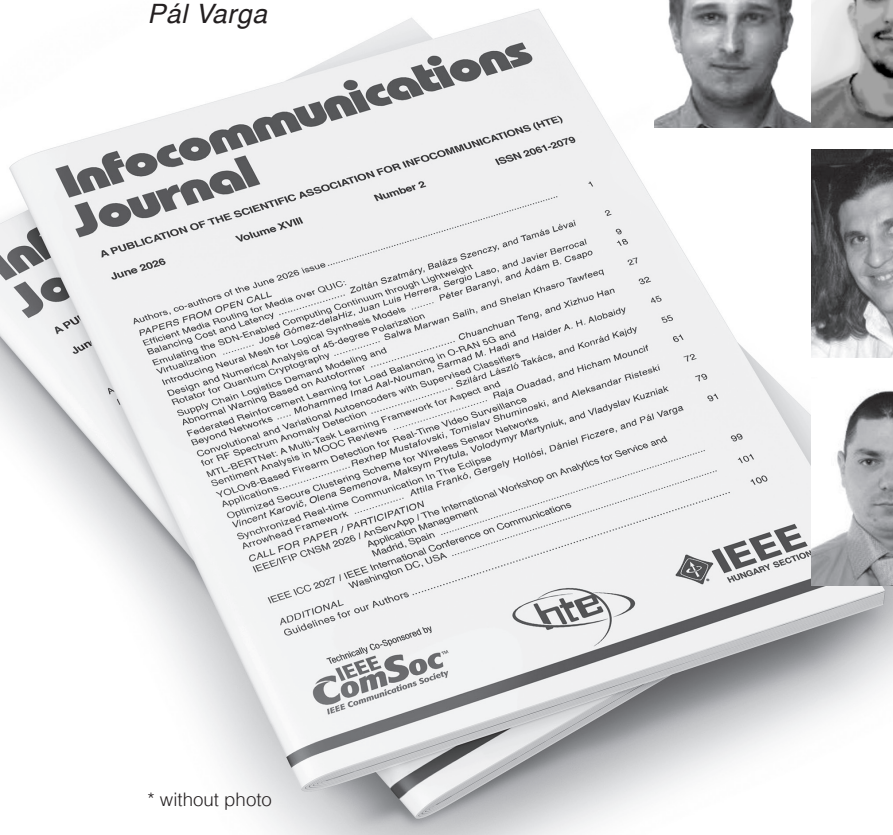
Publisher: PÉTER NAGY

HU ISSN 2061-2079 • Layout: PLAZMA DS • Printed by: FOM Media

www.infocommunications.hu

Authors, co-authors of the June 2026 issue

Zoltán Szatmáry, Balázs Szenczy, Tamás Lévai, José Gómez-delaHiz, Juan Luis Herrera, Sergio Laso, Javier Berrocal, Jaime Galán-Jiménez Péter Baranyi, Ádám B. Csapo, Salwa Marwan Salih, Shelan Khasro Tawfeeq, Chuanchuan Teng, Xizhuo Han*, Mohammed Imad Aal-Nouman, Sarmad M. Hadi, Haider A. H. Alobaidy, Szilárd László Takács, Konrád Kajdy, Raja Ouadad, Hicham Mouncif, Rexhep Mustafovski, Tomislav Shuminoski, Aleksandar Risteski, Vincent Karovič, Olena Semenova, Maksym Prytula, Volodymyr Martyniuk, Vladyslav Kuzniak, Attila Frankó, Gergely Hollósi, Dániel Ficzere, Pál Varga*



* without photo

Efficient Media Routing for Media over QUIC: Balancing Cost and Latency

Zoltán Szatmáry, Balázs Szenczy, and Tamás Lévai

***Abstract*—Media accounts for over 80% of global Internet traffic, much of it being latency-sensitive. The IETF's Media over QUIC (MoQ) protocol draft leverages QUIC for scalable media delivery, using a relay-based architecture similar to CDNs. However, current MoQ implementations rely on full mesh relay networks, leading to high network utilization and scalability issues.**

In this work, we present a novel approach to enforce media routing and lower the overall costs of operating a MoQ relay topology while satisfying delay service level objectives. We formulate the problem mathematically, analyze its complexity, and develop optimal and heuristic algorithms. We integrate our approach with a widely-used MoQ implementation. Our evaluation using different synthetic and realistic topologies shows improvements in cost efficiency while keeping all delay constraints satisfied.

***Index Terms*—Real-time systems, Content delivery networks, Streaming media, Multimedia communication, Routing**

I. Introduction

We use media-based solutions in almost every aspect of our digital lives, from voice and video calls to live streaming, cloud gaming, and beyond. This contributes significantly to the network traffic on today's large-scale networks. Some surveys show that media traffic puts more than 80% of all network traffic [1], therefore, the cost-efficient distribution of large-scale media is crucial for current and future networks.

A major challenge in large-scale media streaming over public networks like the Internet is ensuring liveness (e.g., conversation in a video call) and, in many use cases, enabling real-time interactions (e.g., user inputs in cloud gaming) due to the lack of control over the intermediary devices involved during the transmission of packets from source to destination. Moreover, applications vary in bandwidth requirements, latency constraints, and communication patterns.

Recently, Media over QUIC (MoQ) [2] emerged as a streamlined, low-latency media delivery solution. MoQ supports a variety of use cases and maintains a straightforward architecture that includes publishers, subscriber clients, and relays. Relays are responsible for transmitting the content between the publisher and subscribers. The relay network enables high scalability potential for systems using MoQ. Although MoQ advances quickly, it lacks critical features, such as the media routing capabilities necessary for large-scale and cost-efficient deployments.

Zoltán Szatmáry, Balázs Szenczy, Tamás Lévai are with the HSN Lab, Department of Telecommunications and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary.

Tamás Lévai is also affiliated to HUN-REN-BME Information Systems Research Group.

(correspondence: levait@tmit.bme.hu)

DOI: 10.36244/ICJ.2026.2.1

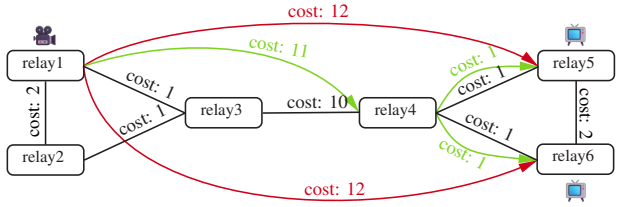


Figure 1. Topology of 6 MoQ relays: publisher connects to relay1, subscribers connect to relay5 and relay6; the naive approach (no media routing) costs 24, while cleverly routed media incurs 12 cost only.

To demonstrate the need for media routing, we present a MoQ network illustrated in Fig. 1. Relays are represented by boxes, with the publisher's relay marked with a camera, and the subscribers' relays marked with a television icon. Overlay connections are indicated by red and green arrows, while black lines represent underlay connections. We use overlay links to demonstrate IP routing, utilizing the same underlay links between two hosts but using OSI Layer 3 routing. In contrast, using underlay links enables routing media at Layer 7, leveraging protocol-defined tracks and groups. Media routing in Layer 7 can effectively utilize the underlay links (e.g., by path aggregation). The cost-benefit of media routing is evident even in this small example: with IP routing, using the red overlay links to send data to the two subscribers results in a cost total of 24. In contrast, with media routing, data is sent only once until `relay4`, where it is subsequently divided for the two subscribers, halving costs to 12 units. It is important to emphasize that in real-world provider networks, many more relays would likely be deployed to accommodate the large demand. However, our intention here is merely to illustrate the general concept of media routing in MoQ and its benefits, which remain clearly visible even in such a simplified network.

Intuitively the spectrum of possible solutions for MoQ media routing consists of two extremes: one minimizing the latency (provided by the state-of-the-art *no media routing* solution) and the other minimizing operational costs (provided by a Minimum Spanning Tree (MST) based optimization strategy that does not consider any delay bounds). In most cases neither of them suits our goals and a balance has to be found between the two. There is a trade-off: we have to compromise on one parameter to gain on the other, so the ideal solution to the problem should be flexible. While the main goal should be minimizing costs, when the delay bound cannot be satisfied via a certain link, we should switch strategies and adapt to the situation by falling back to using another link that may not be

cost-optimal otherwise but conforms to the delay bounds.

Consequently, our goal with this paper is to enable latency-aware media routing in MoQ relay networks. Our contributions in this paper are as follows:

Analytical model. We present our take on modeling and optimizing MoQ relay topologies in §III. We show a mathematical formulation of the relay topology optimization problem.

Heuristics. We present a heuristic algorithm to solve the optimization in polynomial time, making it a good fit for real-life usage (§III).

Design, implementation, and evaluation. We extend the MoQ architecture with media routing capabilities. We present a practical implementation using the `moq-rs` implementation (§IV). In §V, we evaluate our model and heuristics numerically and demonstrate the effectiveness of our solution in a realistic testbed. Our implementation is available for download at [3]. We close the paper discussing related work (§VI) and deriving the main conclusions (§VII).

II. Media Routing in Media over QUIC

Media over QUIC offers a modern approach to media transmission, enhancing scalability and reducing latency by utilizing relays that streamline content distribution. This architecture makes it highly effective for dynamic and large-scale media applications. Yet, we argue that our problem with the cost-efficient and low-latency distribution of content in MoQ is closely related to other (usually video-based) media distribution problems. A ubiquitous problem that arises in virtually every content delivery system is that the centrality of the data has to be given up in order to provide better latency to users. MoQ builds upon the Content Delivery Networks (CDN) design distributing content across multiple servers (relays in MoQ) at several physical locations instead of relying on a single centralized server so that users can get the content from the nearest or otherwise optimal server, enabling low-latency distribution of on-demand content.

To utilize the whole potential of this architecture, MoQ requires media routing to optimize network usage and provide QoS. One well-known example of media routing is multicasting. Point-to-point unicast works well for many use cases; however, in cases where the same content needs to be distributed to multiple peers, this poses unnecessary duplications and thus increased utilization on the network. An early attempt at solving this came with IP Multicast. Although it has become a go-to solution for efficient media distribution on manged networks (e.g., for use in IPTV [4]), due to the lack of global support for IP Multicast protocols, it is unavailable for use on the public Internet [5]. Recognizing these constraints, Application Layer Multicast (ALM) emerged as a practical Layer 7 approach to bring multicast capabilities to networks without native IP Multicast support at Layer 3 [6].

Media routing solutions based on ALM provide the scalability needed for MoQ media distribution in large-scale networks. An ideal solution must also consider end-to-end (E2E) delays from publisher to subscriber to ensure service quality (i.e., QoS). Thus, our solution improves a state-of-the-art MoQ implementation by extending it with media routing capabilities to provide ALM and guarantee QoS.

MoQ relays are generic and contain no application logic by design [2]. Therefore, the media routing implementation requires an additional component (e.g., a routing API server) in the system that is aware of the QoS requirements, and can route media requests through the relay network accordingly. Take Fig. 1 as an example, subscriber at `relay6` subscribes to the track originally published at `relay1`. The router, overseeing the network, can detect that this content is already served to `relay5` via `relay4`, which is closer to `relay6`, so its routes `relay6` subscription to `relay4` thus *iteratively builds a distribution network for the track*.

By default, all relays in a network have one common API server to which they connect, and all clients know their next-hop relays. Before a publisher begins serving a track, its relay notifies this API server that the track will be available. When a subscription arrives at a relay, it checks whether the track is already available either locally (i.e., the publisher is directly connected) or remotely (i.e., an upstream subscription is already in place to a relay providing the track). If available, the relay acknowledges the subscription and begins forwarding the track as its objects arrive. Otherwise, it queries the API server to determine the appropriate relay to request the track from. The API server acts as a centralized controller, similar to a Software-Defined Networking (SDN) controller, and coordinates the creation of media forwarding paths through subscriptions between relays. Internally, it may use any method for path determination.

For example, `moq-api`, the API server of `moq-rs`, connects all relays in a full mesh topology. As a result, after consulting the API server, a subscriber's relay is always instructed to request the track directly from the publisher's relay. While this achieves distribution with minimal latency overhead, this method is not scalable.

Media routing decisions are made per relay, so the API server must know which relay is making the request to respond correctly. Since this information is not currently sent to the `moq-rs` API server, we modified the relays to address this limitation (§IV).

III. Modeling and Optimizing The Relay Topology

Media routing essentially defines a topology for each track, specifying the next-hop relay (i.e., the relay from which to request the track) for each relay involved in forwarding.

This section outlines the formalization of our media routing problem and describes various optimization methods (e.g., heuristic and ILP-based) that can be used by a media routing algorithm to determine the optimal topology, comparing their benefits and potential trade-offs.

Our take on modeling network topologies formed by relays resembles key characteristics of CDNs. We define relay topologies (or networks) as special graphs where the nodes represent relays and the links represent logical interconnections (over Layer 7) between each two nodes of the graph. These interconnections leverage the end-to-end connectivity provided by lower-level layers; therefore, when modeling links between relay nodes, a physical path (e.g., the shortest path) going through a series of routers and other network devices between two nodes is relaxed as a single logical link.

In our model, we abstract the fact that in MoQ, (original) publishers and (end) subscribers are not actually relays, but simple end devices that connect to (possibly to the nearest) relays. We can do this without loss of generality, since it is not a significant difference to real life. For this reason, we will refer to publishers and subscribers as their relays, in contrast to the terminology used in the IETF draft [2].

The media content in our model is organized into tracks. We strive to make an optimal topology for each track. However, for each track we only account for the potential traffic generated by the distribution of itself, which means that even when there are multiple tracks present, each of those will have a topology optimized independently of each other in complete isolation. This relaxation is acceptable under our conditions, due to the fact that we are working with logical links, of which there may be many possible mappings to disjoint paths of physical links; therefore, two independent tracks sharing the same logical link will likely not result in bandwidth constraints due to massive overprovisioning. Furthermore, studies have shown that while the capacity of certain fiber-optic systems is starting to saturate, proposed techniques to overcome this limitation [7] already exist [8].

A. The Optimization Problem

Unlike prior CDN cost models that optimize replica placement or request routing over a fixed server infrastructure [9], or ALM solutions that construct multicast trees in a distributed, one-shot manner [10], [11], our formulation incrementally builds per-track media forwarding trees as subscribers join, enforcing E2E delay bounds within MoQ's relay-based pub/sub model. Specifically, the goal is to establish an interconnection among relays between a track's publisher and its subscribers, minimizing the total operational cost while satisfying delay requirements. The optimization takes a relay network and one or more tracks as input. It produces an optimal forwarding topology for each track.

The relay network is given as a directed graph $G(V, E)$, with $v \in V$ being the relays and $e \in E$ the links between them. Each link carries a cost c_{ij} and a delay d_{ij} for $(i, j) \in E$. The cost is an abstract value encoding the relative preference for using a link as higher cost indicates a link the optimizer should avoid unless necessary. Delay values are physically measurable quantities, though in practice they may be approximations of the true path delay (e.g., based on propagation delay only).

Each track t is given with four attributes: its publisher denoted by $P^t \in V$, its set of subscribers $S^t \subseteq V$, a mapping of roles R_f, and a delay bound $D^t \in \mathbb{N}$ assigned to it.

The roles of the publisher and the subscriber in a track are formalized by streams (or flows) indicated by $f \in F^t$, each with a path from the publisher of a subscriber across G, and with a vector R_f which encodes roles of the relays with regard to that flow. Therefore, for each $f \in F$ and $i \in V$, R_i^f is set to -1, if $i \in P^t$, +1, if $i \in S^t$, or 0 otherwise.

The delay bound is the maximum tolerated E2E delay between the publisher and each subscriber, where the E2E delay is measured independently for each subscriber by adding the delay values along the path between the publisher and the subscriber in the topology.

To measure link utilization, we introduce two other variables: $x_{ij}^f : (i, j) \in E, f \in F^t$ which represents the theoretical "transmission bitrate" (measured in units) of stream f on link (i, j); and $y_{ij}^t : (i, j) \in E$ that represents the link usage of the track on (i, j) for each $t \in T$. To include the delay bound, we add an additional variable $z_{ij}^f \in \{0, 1\}$ to indicate whether the link (i, j) is on the path of stream $f \in F^t : \forall t \in T$.

The objective function is characterized as the sum of link costs in the topology for each track $t \in T$: $\sum_{t \in T} \sum_{e \in E_t} c(e)$. The solution is an optimal topology $G_t(V_t, E_t) : V_t \subseteq V, E_t \subseteq E$ for each track $t \in T$. A topology is optimal if *1)* the operation of the topology is cost-optimal and minimizes extra network usage, and *2)* meets the E2E delay requirements.

Putting it all together, the Integer Linear Program that satisfies these requirements is the following:

$$
\begin{aligned}
\min \quad & \sum_{t \in T} \sum_{(i,j) \in E} c_{ij} y_{ij}^t & \\
& y_{ij}^t \geq x_{ij}^f & \forall (i,j) \in E, \forall f \in F^t : \forall t \in T \\
& \sum_{j:j,i \in E} x_{ji}^f - \sum_{j:i,j \in E} x_{ij}^f = R_i^f & \forall i \in V, \forall f \in F^t : \forall t \in T \\
& x_{ij}^f \geq 0 & \forall (i,j) \in E, \forall f \in F^t : \forall t \in T \\
& y_{ij}^t \geq 0 & \forall (i,j) \in E, \forall t \in T \\
& z_{ij}^f * M \geq x_{ij}^f & \forall (i,j) \in E, \forall f \in F^t : \forall t \in T \\
& \sum_{i,j \in E} z_{ij}^f * d_{ij} \leq D^t & \forall f \in F^t : \forall t \in T
\end{aligned}
$$

This problem is referred to as the Bounded Diameter Minimum Steiner Tree (or Bounded Diameter Steiner Minimum Tree [12]) problem [13], which is known to be NP-hard.

B. Heuristics

The ILP-based solution is complex to solve. It is NP-hard (see [14]), and for a considerably small network with only 5 to 10 active relays its execution time takes up minutes, making it limited for real deployments.

Due to the long processing times, we propose a simple heuristic approximation to find close-to-optimal routes for media streams over the relay network. We opted for a dynamic approach that does not require full recomputation of the topology when new subscribers join. This design enhances real-world applicability and potentially improves computational speed. We employed an algorithm that follows a greedy expansion strategy with local refinements leading to close-to-optimal solutions with modest computational overhead.

The full mesh nature of the input graph allowed us to leverage the constrained MST relaxation [10], which enables to efficiently compute (locally) optimal connections to the distribution tree for new subscribers. The intuition behind the algorithm is simple and can generally be described with two main ideas. *1)* For each subscriber, we try to find a link that connects the subscriber to the existing distribution tree in a way that satisfies the delay constraints but is the least expensive among such links. If such a link exists, use it to create the new subscriber to the tree; otherwise, the problem is infeasible. *2)* See if there is a way to reduce costs by reaching another subscriber with a link that goes from this newly added

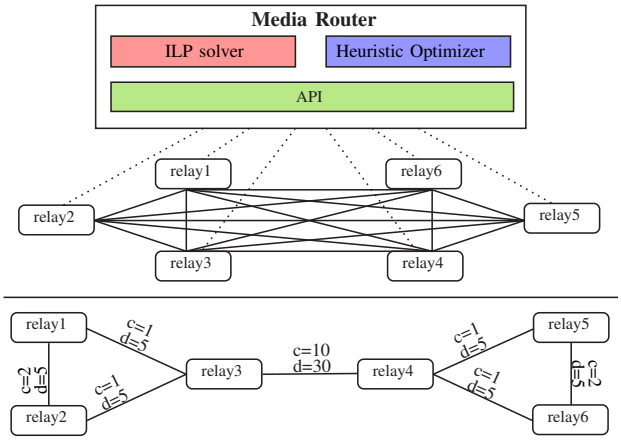


Figure 2. Architecture: optimizer and the relay network (top) with the underlay network (bottom).

subscriber. If there is, choose the one with the lowest cost and the least E2E delay.

Notice how this algorithm can be easily converted to work with dynamically added subscribers. We call this the *Heuristic Media Routing* in later sections. The total complexity of the heuristic algorithm is $O(n^3)$.

IV. Implementation

Based on our theoretical findings, we extended MoQ with media routing capabilities that make media distribution cost-efficient and latency-aware. To confirm our results and assess the feasibility of using our model as a media routing extension to MoQ, we devised two interdependent solutions. Firstly, we created an optimizer prototype that implements the integer linear program and the *Heuristic Media Routing* algorithm described in §III. Secondly, we enhanced `moq-rs` to enable support for dynamic topology construction. Next, we will outline the high-level architecture along with the primary design decisions considered. The source code is publicly available on GitHub [3].

A. Architecture

We strove to make all control in our system centralized, so that relays are fully coordinated. In Figure 2, we can see how the Media Router acts as a single source of truth controlling every relay in the system, resembling the widely-used SDN architecture [15].

The Media Router component consists of two subcomponents: the topology optimizer and the API that provides access to the former through HTTP endpoints.

The API server shares the same endpoints as the upstream `moq-api`. When the relay consults the API (e.g., upon a new subscription to a track that is not currently available at the relay), instead of using the original `moq-api`, an endpoint on the Media Router's API subcomponent is invoked, triggering an immediate recalculation of the optimal topology upon request.

The Topology Optimizer determines the topology for distributing a specified track throughout the network. It takes as input a network containing all system relays, along with the track for which the topology needs to be optimized. The Topology Optimizer employs multiple strategies, including the state-of-the-art No Media Routing solution, an Integer Linear Program solver, the *Heuristic Media Routing*, and a Minimum Spanning Tree-based approach to find an optimal or a near-optimal topology.

V. Evaluation

Our evaluation goals were twofold: *1)* to compare the theoretical complexities of the different solvers presented to their empirical performance; and *2)* to demonstrate the usability of our system with realistic components. We test our theoretical solutions on a benchmark to see how they scale in a simulated environment. Then, we use a Mininet-based testbed to see how the system behaves when using realistic components. Our evaluation tools are available on GitHub [3].

We used a computer with an AMD EPYC 7502P (32 CPU cores) and 128 GB RAM, running Ubuntu 22.04 with Python 3.10 and Rust 1.77.2 installed. We switched to performance mode and fine-tuned network configuration parameters.

A. Benchmarks (Numerical Evaluation)

For numerical evaluation of the media routing algorithms, we construct a large-scale realistic network we call *Azure-GÉANT*: we place MoQ relays at various Microsoft Azure Regions worldwide using approximate geographical locations [16], and we connect these relays via the links of the GÉANT research network to simulate the underlay of a large-scale network [17]. The GÉANT map consists of cities connected by links grouped into regions. We leveraged this for cost assignments: interregional links received a cost equal to the total number of cities in the network, while intraregional links had a cost of 1. To map the underlay to the overlay, we first linked each relay to its nearest city with zero-cost "last-mile" connections. Next, we formed a full-mesh overlay by connecting each unique relay pair through the shortest path between their corresponding cities, assigning delay values based on straight-line distance and costs by summing the costs of the underlay links used.

To compare the complexity of the different optimizers at scale, we performed simulations on *Azure-GÉANT*. In each simulation run, we have measured the total time taken to run the optimization process, as well as the total cost and the maximum E2E latency present in the resulting topology. The optimizers included in the benchmarks were: (1) No Media Routing (baseline, the state of the art); (2) Heuristic Media Routing; (3) Integer Linear Program (ILP); and (4) Minimum Spanning Tree (MST). Optimizers (2) and (3) have been well described in §III, (1) and (4), on the other hand, are implementations corresponding to the two extreme points of the delay-cost spectrum introduced in §I which we have included for comparison.

We have set up the simulation with a total of 40 peers geographically located in multiple continents, of which an ever-increasing subset is selected for the track (with the first

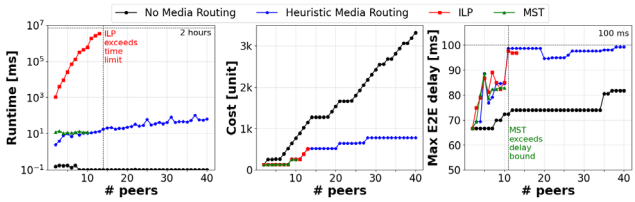


Figure 3. Numerical Evaluation: Optimizer performance metrics for a single track with 100 ms delay bound over an increasing number of peers.

peer included in the subset being the publisher and every other a subscriber) in each test run of the simulation.

In Figure 3, we can see the results as the number of peers increases. Note that for better visualization of orders of magnitude, we have plotted the runtime values on a logarithmic scale. When an optimizer cannot produce a suitable topology adhering to the delay bound within a 2 hour time limit, the plotting of that optimizer ends. We have denoted these breakpoints with vertical lines, and specified rationales.

No Media Routing is the state of the art for MoQ implementations. Compared to other algorithms, it is a simple algorithm that provides the best delays on the cost of massive resource usage (4× increment in costs with 40 peers). MST cannot satisfy the 100 ms delay bound when the number of peers exceeds 11, which confirms that naively optimizing for costs may not always be sufficient. The ILP follows an exponential runtime curve due to its NP-completeness, and just after 14 peers, its optimization process could not finish in a 2 hour time frame. Conversely, the Heuristic Media Routing produces outputs well within the time limit thanks to its $O(n^3)$ algorithmic complexity. It can effectively minimize costs while ensuring the delay bound.

A staircase pattern, observable in all optimizers, arises from the discrete jump in both cost and delay when a subscriber is reached via an inter-regional link rather than an intra-regional one. This is a structural property of the *Azure-GÉANT* topology, where inter-regional costs are set to N (total city count), rather than 1 for intra-regional links. This figure also shows that the idea of the delay-cost spectrum (§I) is present in practical scenarios as well.

B. Real-life Measurement in a Testbed

To demonstrate the usability of MoQ media routing and our optimizer, we recreated the network of Fig. 1 using the Mininet [15] network emulator. We use our `moq-rs` version (recall §IV). The *Original* method is the current no media routing behavior, while the *Optimized* includes the media route optimizer. The optimizer gives similar routes as in Fig. 1. Our measurements focused on two metrics: end-to-end (E2E) delay and network usage. Table I shows the results of 5 consecutive runs. We see a slight increase in E2E delays due to the extra relay in the path. The benefit of media routing is apparent in the network usage of the relay network. With the optimization, the system transmits 13.5 MB data instead of 40.7 MB (appr. 67% saved).

To conclude our evaluation, the results show that our media routing algorithm scales well. Optimized media routing can reduce link usage effectively. Still, in some cases, it might add minimal delays, making them a practical choice for scalable media routing for MoQ.

TABLE I
TESTBED: PERFORMANCE COMPARISON OF THE NO MEDIA ROUTING (ORIGINAL) AND THE LATENCY-AWARE MEDIA ROUTING (OPTIMIZED) CONFIGURATIONS IN THE NETWORK OF FIG. 1: SLIGHT EXTRA DELAY DUE TO THE EXTRA RELAY, BUT SIGNIFICANT SAVING IN RESOURCE USAGE BY PATH AGGREGATION.

Metric	Method	Mean	Min	Max
End-to-end delay [ms]	Original	45.65	45.50	45.852
	Optimized	**48.94**	48.72	49.117
Network usage [MB]	Original	40.73	40.56	40.876
	Optimized	**13.56**	13.52	13.60

C. Limitations and Scope

The current formulation optimizes each track independently, which is well-founded when logical links map to disjoint physical paths (a reasonable assumption in overprovisioned backbone networks). However, in congested deployments where multiple high-bandwidth tracks compete for the same physical bottleneck, per-track independence may underestimate resource contention, requiring joint multi-track optimization. This is a natural extension of the present work.

The testbed evaluation is intentionally limited to the six-relay topology shown in Fig. 2, which clearly illustrates the core cost-saving mechanism. The numerical benchmark on *Azure-GÉANT* validates scalability up to 40 peers per track; evaluating concurrent multi-track scenarios at that scale is future work. We note, however, that the per-track heuristic's $O(n^3)$ complexity holds regardless of the number of tracks, so the computational argument for real-world deployability is unaffected.

Finally, the heuristic is designed for incremental subscriber arrivals and does not perform a full topology recomputation due to churn. In highly dynamic settings with rapid subscriber joins and leaves, the maintained topology may degrade over time. Periodic reoptimization or churn-aware extensions are straightforward directions that we leave for future work.

VI. RELATED WORK

Media over QUIC offers a modern approach to media transmission, enhancing scalability and reducing latency by using relays that streamline content distribution, making it well-suited for dynamic, large-scale media applications [18], [19] and beyond [20]. The importance of relay topology design for MoQ has been recognized within the IETF itself [21], and standardized relay benchmarking frameworks are an emerging direction [22]. Before MoQ, prior solutions included HTTP Live Streaming (HLS), Dynamic Adaptive Streaming over HTTP (DASH) [23], [24], WebRTC [25], and Real-Time Messaging Protocol (RTMP) [26]. Each of these focused on a subset of media transmission use cases; none of them can be applied universally. For example, while WebRTC is the prevailing solution for delay-sensitive, live media transmission over the Internet, it is commonly criticized for being difficult

to scale and adapt to application-specific requirements. This is because QoS-affecting parameters, such as behavior under packet loss or insufficient throughput, are mostly baked into the protocol and the browser implementations. HLS and DASH, in contrast, rely on HTTP's request-response model, in which clients explicitly request individual media segments. While this approach scales well for static content distribution, such as Video on Demand, it introduces latency that limits its suitability for interactive applications. Finally, RTMP is primarily designed for media contribution from publishers to streaming servers rather than for large-scale content distribution. MoQ aims to become a unifying standard covering the use cases of all of these. While MoQ represents a novel integration of multiple techniques, many of its underlying concepts have been explored in earlier work. Our problem with the cost-efficient and low-latency distribution of content in MoQ is closely related to existing media distribution problems. Next, we survey prior work that motivates and contextualizes our approach.

The general principle of content delivery systems is to give up on the centrality of data to achieve better latency for users [27]. Content Delivery Networks (CDNs) work by distributing content across multiple servers at several physical locations, instead of relying on a single centralized server, so that users can access content from the nearest or otherwise optimal server. The main benefits of CDNs are caching, cost-efficient operation (due to only having to move large volumes of data once) and low-latency distribution of content [9].

Another relevant approach is multicasting, which involves simultaneous transmission of the same data to one or multiple destinations [28]. Media delivery is, in fact, a particularly common application of multicasting. A prime example of media multicasting is IPTV [4]. The main problem with multicasting, however, is that it has no direct support in the Internet Protocol, and therefore, implementing Layer 3 multicasting in IP networks requires specialized protocols such as Protocol Independent Multicast and Internet Group Management Protocol. Thus, multicast is mostly limited to private, managed networks with appropriate equipment supporting the required protocols and configured accordingly (e.g., telco providers' IPTV networks).

Consequently, neither CDN nor IP Multicast is sufficient on its own for live media distribution over the Internet since Content Delivery Networks are tailored for Video on Demand use cases, and IP Multicast is constrained to managed networks.

Before our work, numerous studies proposed shifting multicasting from the network layer to the application layer, Application Layer Multicasting (ALM), in which multicast topologies are constructed as overlay networks atop large-scale infrastructures such as the Internet. A notable example is peer-to-peer (P2P) streaming networks, which have traditionally relied on ALM by design. One well-known implementation of P2P streaming is the Narada protocol, which realizes the End System Multicast (ESM) architecture [29]. Narada constructs the overlay network in a self-organizing manner, using network metrics obtained through periodic heartbeat measurements between end systems participating in multicast content delivery.

In contrast, other approaches such as Topology Aware Grouping (TAG) [11] incorporate knowledge of the underlying physical network into overlay multicast routing. Although such topology information is not always available, it proves highly valuable when it is. Without it, peers participating in media delivery must conduct their measurements to infer the network topology, which limits scalability. Therefore, this ability to leverage existing topology information is a key differentiator between TAG and ESM.

A solution combining the strengths of CDNs and ALMs has the potential to provide an efficient distribution model for live media on Internet-scale networks. For this reason, we built on top of the idea of topology-aware overlay networks. Still, our goals were different regarding how and where the routing should occur. For example, we aimed for centralized control, similar to SDN networks, rather than the inherently distributed TAG, ESM, and other P2P streaming solutions. Second, while TAG is core-based, our extension of the MoQ API server ensures that relays contact only their next-hop relays within a single track. Finally, many of the aforementioned ALM solutions explicitly tie topology construction to a specific metric, such as bandwidth; however, our solution can use any cost function as long as it satisfies the triangle inequality.

More recent works have applied similar ALM ideas to WebRTC-based systems. For example, Silva *et al.* propose a hybrid architecture combining P2P multicast trees with a centralized control plane [30], foreshadowing the SDN-inspired centralized routing approach we adopt for MoQ.

VII. CONCLUSIONS

Media over QUIC is a novel initiative for large-scale media distribution, which uses a network of relays for transmission. Current implementations use full mesh relay topology, leading to high network utilization and limited scalability.

In this paper, we presented a novel approach to enforce media routing by optimizing the MoQ relay topology based on service delay requirements. Our solution lowered the overall costs and enabled high scalability. We formulated the media routing optimization problem and introduced heuristic routing algorithms. We integrated our approach with a widely-used MoQ implementation. We demonstrated that our solution improves cost-efficiency while meeting service delay bounds in synthetic and realistic environments.

Future work will address joint multi-track optimization, larger-scale multi-stream evaluation, and churn-aware heuristic extensions.

ACKNOWLEDGMENT

Special thanks to Gábor Rétvári, Felicián Németh, István Pelle, and Gergely Pongrácz for the discussions that led to this line of work. We thank Ericsson Hungary for providing the infrastructure to run our measurements. We also express our gratitude to Luke Curley and the supportive `moq-rs` community.

REFERENCES

[1] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach (8th Edition)*, 8th ed. London, UK: Pearson, 2020.

[2] S. Nandakumar, V. Vasiliev, I. Swett, and A. Frindell, "Media over QUIC Transport," Internet Engineering Task Force, Internet-Draft draft-ietf-moq-transport-11, Apr. 2025. [Online]. Available: https://datatracker.ietf.org/doc/draft-ietf-moq-transport/11/

[3] HSN Lab, "Github repository: Optimizer and testbed," https://github.com/hsnlab/moq-relay-topo-opt, 2025.

[4] S. Zeadally, H. Moustafa, and F. Siddiqui, "Internet protocol television (iptv): Architecture, trends, and challenges," *IEEE Systems Journal*, vol. 5, no. 4, pp. 518–527, 2011.

[5] C. Diot, B. N. Levine, B. Lyles, H. Kassem, and D. Balensiefen, "Deployment issues for the ip multicast service and architecture," *IEEE network*, vol. 14, no. 1, pp. 78–88, 2000.

[6] M. Hosseini, D. T. Ahmed, S. Shirmohammadi, and N. D. Georganas, "A survey of application-layer multicast protocols," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 3, pp. 58–74, 2007.

[7] C. Xie and B. Zhang, "Scaling optical interconnects for hyperscale data center networks," *Proceedings of the IEEE*, vol. 110, no. 11, pp. 1699–1713, 2022.

[8] G. Rademacher, B. J. Puttnam, R. S. Luís, J. Sakaguchi, W. Klaus, T. A. Eriksson, Y. Awaji, T. Hayashi, T. Nagashima, T. Nakanishi *et al.*, "10.66 peta-bit/s transmission over a 38-core-three-mode fiber," in *optical fiber communication conference*. Optica Publishing Group, 2020, pp. Th3H–1.

[9] B. Zolfaghari, G. Srivastava, S. Roy, H. R. Nemati, F. Afghah, T. Koshiba, A. Razi, K. Bibak, P. Mitra, and B. K. Rai, "Content delivery networks: State of the art, trends, and future roadmap," *ACM Comput. Surv.*, vol. 53, no. 2, Apr. 2020.

[10] Q. Zhu, M. Parsa, and J. Garcia-Luna-Aceves, "A source-based algorithm for delay-constrained minimum-cost multicasting," in *Proceedings of INFOCOM'95*. New York, NY, USA: IEEE, 1995, pp. 377–385 vol.1.

[11] M. Kwon and S. Fahmy, "Topology-aware overlay networks for group communication," in *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*, 2002, pp. 127–136.

[12] K.-H. Vik, P. Halvorsen, and C. Griwodz, "Evaluating steiner-tree heuristics and diameter variations for application layer multicast," *Computer Networks*, vol. 52, no. 15, pp. 2872–2893, 2008.

[13] A. Mashreghi and A. Zarei, "When diameter matters: parameterized approximation algorithms for bounded diameter minimum steiner tree problem," *Theory of Computing Systems*, vol. 58, no. 2, pp. 287–303, 2016.

[14] R. M. Karp, *Reducibility among Combinatorial Problems*. Boston, MA: Springer US, 1972, pp. 85–103.

[15] B. Lantz, B. Heller, and N. McKeown, "A network in a laptop: rapid prototyping for software-defined networks," in *Proceedings of the 9th ACM SIGCOMM HotNets*. New York, NY, USA: ACM, 2010, pp. 1–6.

[16] Microsoft, "Microsoft datacenters," https://datacenters.microsoft.com/globe/explore, 2025, accessed: 2025-05-04.

[17] G. Association, "GÉant connectivity map," https://map.geant.org/, 2024, accessed: 2024-10-31.

[18] Z. Gurel, T. Erkilic Civelek, A. Bodur, S. Bilgin, D. Yeniceri, and A. C. Begen, "Media over quic: Initial testing, findings and results," in *Proceedings of the 14th ACM Multimedia Systems Conference*, ser. MMSys '23. New York, NY, USA: ACM, 2023, p. 301–306.

[19] A. C. Freeman, "Scalable event-based video streaming for machines with moq," in *Proceedings of the 4th Mile-High Video Conference*, ser. MHV'25. New York, NY, USA: ACM, 2025, p. 60–66.

[20] S. Nandakumar, "SPACE - Scalable Pubsub, Asymmetric and CachEd transport for Deep Space communications," Internet Engineering Task Force, Internet-Draft draft-nandakumar-deepspace-moq-00, Oct. 2024, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/draft-nandakumar-deepspace-moq/00/

[21] H. Shi *et al.*, "Design Space Analysis of MoQ," IETF, Tech. Rep., Oct. 2022, internet-Draft draft-shi-moq-design-space-analysis-of-moq-00.

[22] T. Evens, T. Rigaux, and S. Henning, "Media over QUIC Relay Benchmark Methodology," *IETF, Tech. Rep.*, Nov. 2025, internet-Draft draft-evens-moq-bench-00.

[23] R. Pantos and W. May, "HTTP Live Streaming," RFC 8216, Aug. 2017. [Online]. Available: https://www.rfc-editor.org/info/rfc8216

[24] International Organization for Standardization, Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats, ISO/IEC Std. 23009-1:2014, May 2014.

[25] N. Blum, S. Lachapelle, and H. Alvestrand, "Webrtc: real-time communication for the open web platform," *Commun. ACM*, vol. 64, no. 8, pp. 50–54, Jul. 2021. [Online]. Available: DOI: 10.1145/3453182

[26] H. Parmar and M. Thornburgh, "Adobe's real-time messaging protocol (rtmp)," Adobe Systems Incorporated, Tech. Rep., 2012.

[27] A. Popescu, D. D. Kouvatsos, D. Remondo, and S. Giordano, "Content distribution over ip: developments and challenges," in *Network Performance Engineering: A Handbook on Convergent Multi-Service Networks and Next Generation Internet*. Heidelberg, DE: Springer, 2011, pp. 979–987.

[28] L. Peterson and B. Davie, "Computer networks: A systems approach - multicast," https://book.systemsapproach.org/scaling/multicast.html, 2025, accessed: 2025-05-07.

[29] Y.-h. Chu, S. G. Rao, S. Seshan, and H. Zhang, "A case for end system multicast," *IEEE Journal on selected areas in communications*, vol. 20, no. 8, pp. 1456–1471, 2002.

[30] M. A. Silva, R. Bertone, and J. Schäfer, "Topology distribution for video-conferencing applications," in *2019 10th International Conference on Networks of the Future (NoF)*. IEEE, 2019, pp. 90–97.

Zoltán Szatmáry is an MSc student in Computer Engineering at the Budapest University of Technology and Economics (BME). His research interests include network softwarization, data center networking, and IT security.

Balázs Szenczy has received his MSc degree in Computer Engineering from the Budapest University of Technology and Economics (BME) in January 2025. He currently serves as a DevOps Engineer at Nokia. His professional focus includes cloud-native networking solutions, containerization, and automated deployment pipelines. While maintaining an industry position, he actively follows developments of Media over QUIC and its community.

Tamás Lévai received the MSc in Computer Engineering at the Budapest University of Technology and Economics (BME) in 2016. He received his PhD degree in Informatics in 2022 at BME. Currently, he is an Assistant Professor at BME, and a Research Fellow at HUN-REN TKI. His research interest focuses on computer networks and distributed computing, mainly software-defined networking, cloud native computing and real-time communications.

Emulating the SDN-Enabled Computing Continuum through Lightweight Virtualization

José Gómez-delaHiz* *Student Member, IEEE*, Juan Luis Herrera†, Sergio Laso*, Javier Berrocal* *Member, IEEE*, Jaime Galán-Jiménez* *Member, IEEE*

***Abstract*—In recent years, the number of Internet-connected devices has increased notably, leading to a significant rise in data traffic. This increase has been fostered by the Internet of Things paradigm, the use of microservices architectures in application development, and the ability to deploy these applications across various layers in the Computing Continuum (including Fog, Edge, and Cloud layers). Consequently, choosing the right deployment strategy has become essential for network operators and developers, especially in intensive domains such as smart cities. In this work, we introduce an emulation framework that allows developers and operators to decide how to deploy networks, computing devices, and applications in a Computing Continuum environment, ensuring compliance with established Quality of Service standards. This framework supports both IP and SDN network paradigms and is highly adaptable to different scenarios due to its use of container-based virtualization. Furthermore, the SDN paradigm provides flexibility, enabling the implementation of a service discovery feature that simplifies communication between end devices and services. Evaluations conducted in a realistic smart city scenario show that this framework can be extended and applied to a wide range of situations and configurations, meeting the needs of the research community in the Computing Continuum domain.**

***Index Terms*—Computing Continuum, SDN, Fog, IoT, Service Discovery**

I. INTRODUCTION

The Internet of Things (IoT) is a rapidly evolving technology with substantial technical, social, and economic importance. Nowadays, IoT devices are frequently used in diverse environments to carry out daily tasks. In 2023, the number of devices connected to IP networks is approximately 29.3 billion [1]. The emergence of new applications that demand strict Quality of Service (QoS) levels, especially in smart cities, poses a challenge for network operators in managing large volumes of data. In an IoT-based network scenario, optimizing the QoS involves considering three layers: the network layer, the computing layer, and the applications and services layer.

The application and services layer pertains to IoT applications designed using the Microservices Architecture (MSA) pattern. MSA applications are characterized by their division into small, loosely-coupled microservices [2]. When simple tasks work together, they can manage more complex functionalities. The sequence of ordered microservices that make up each application functionality is referred to as a workflow, with each user request linked to a specific workflow [3]. MSAs can be deployed across distributed infrastructure, providing higher flexibility [2]. The problem of optimally placing a set of microservices within a Computing Continuum infrastructure, paramount to optimize application QoS, is referred to in the literature as the Decentralized Computation Distribution Problem (DCDP) [3]. The Computing Continuum is a distributed computing paradigm that extends the cloud with user-proximate computing resources in the fog and edge layers, providing heterogeneous computing capabilities across the network to reduce latency, improve resilience, and decrease energy consumption in distributed applications [4].

The computing layer emphasizes the efficient distribution of services. Traditionally, cloud computing was used to deploy IoT applications. However, since cloud servers are often distant from IoT devices, it is beneficial to shift microservices to the Computing Continuum, particularly to nodes nearer to users, to decrease application response time [3]. Consequently, the placement of Computing Continuum nodes becomes vital for optimizing the QoS. The challenge of optimally positioning these nodes was initially addressed in the context of the fog layer and is referred to as the Fog Node Placement Problem (FNPP) [5].

Finally, the network layer manages communications. The adoption of technologies like Software-Defined Networks (SDN) and Network Function Virtualization (NFV) aligns well with the concept of decentralizing IoT applications through the MSA paradigm [6]. Therefore, the Controller Placement Problem (CPP) [7] is a critical challenge in this context, focusing on positioning the SDN controller to ensure optimal QoS for the switches.

Developers of intensive IoT applications are interested in reducing the deployment cost and improving the network performance to meet their QoS targets. To achieve such goals, different authors have proposed methods to solve the CPP, DCDP, and FNPP [3], [5]–[7]. However, to properly evaluate how these methods affect each of the three layers, developers need to execute their software over realistic scenarios. To the best of our knowledge, existing solutions are unable to

* Dept. of Computer Systems and Telematics Engineering, University of Extremadura, Spain
† Distributed Systems Group, Technische Universität Wien, Vienna, Austria
(E-mail: jagomezdh@unex.es, j.gonzalez@dsg.tuwien.ac.at, slasom@unex.es, jberolm@unex.es, jaime@unex.es)
Corresponding author: e-mail: jagomezdh@unex.es

DOI: 10.36244/ICJ.2026.2.2

consider all three layers, are simulated testbeds that cannot be integrated with real application and network software, are tied to a specific method or solver, or need to be configured from scratch in each execution [8]–[10].

To address this need, this work presents an emulation framework, named Emulation Framework for the Computing Continuum (EFCC), to automate the configuration and generation of emulated Computing Continuum testbeds in all three layers. Researchers can use EFCC to evaluate their solutions, ad-hoc or using different methods, for the CPP, DCDP, FNPP or any combination of them. Moreover, as the implementation of service discovery is considered a key feature for applications in the SDN-enabled Computing Continuum [3], EFCC provides built-in support for transparent, network-level service discovery. The main contributions of this work are:

- The proposal of EFCC, a framework for emulating scenarios involving the deployment of MSA-based IoT applications within SDN-based Computing Continuum environments.
- The implementation of transparent service discovery in the SDN controller. EFCC can automatically perform service discovery at the network level through workflow information processing without affecting traditional network traffic, and with minimal effects on QoS.
- Proposal of the DADOSIM scenario modeling format, and the generation of a DADOSIM dataset with scenarios involving different sizes, devices, microservices, and workflows.
- The evaluation of EFCC, including the emulation capabilities, the transparent service discovery functionality, over diverse realistic smart cities-based IoT scenarios, and an unified emulation across cloud-edge-IoT.
- A scalability analysis of emulation performance in Kathará, including the deployment time and the memory usage for each of the tests.

The remainder of this paper is structured as follows. Section II introduces the existing solutions and their limitations. Section III describes the proposed framework and the modules it is composed of. Section IV analyzes the obtained results over realistic use cases. Finally, Section V concludes the work.

II. Related Work

In the Computing Continuum environment, several solutions offer valuable contributions at different scales. These include extensive simulators and emulators for the cloud, fog, and edge layers, which aid in decision-making and provide critical control information about application performance and network infrastructure management. This section reviews these simulators and emulators for the Computing Continuum.

The first work analyzed is FogSim [8], a Java-based simulator for the Fog layer. FogSim evaluates two EDF-based resource management strategies: Distributed EDF for Fog (DEF) and Profit-aware Distributed EDF for Fog (PDEF). DEF improves resource allocation using linear programming while considering latency, processing cost, storage capacity, and energy consumption, whereas PDEF balances dynamic pricing and demand to optimize cost efficiency. Although FogSim enables the analysis of Fog computing scenarios under these strategies, it is tightly coupled to them, requiring explicit implementation to evaluate alternative approaches.

Next, we analyze CloudSim [9], a simulation framework for cloud computing environments. CloudSim allows the configuration of diverse scenarios with different resource settings, allocation policies, and workloads, and supports the evaluation of metrics such as response time, throughput, power consumption, and resource utilization. While this flexibility enables detailed performance analysis and optimization of cloud resource management, CloudSim is limited to traditional cloud computing, lacking support for the Computing Continuum and distributed layers. Moreover, it only supports conventional IP networks, preventing the simulation of SDN-based scenarios, and, as a simulator, shares the same application execution limitations as FogSim.

Another Computing Continuum simulation proposal is YAFS (Yet Another Fog Simulator) [10], a Python-based simulator for Fog and Edge environments. YAFS supports the modelling of Fog infrastructures, including Edge devices, computing resources, and network elements, and allows the evaluation of applications, workloads, and resource management policies through a flexible API for defining topologies and resource configurations. It enables the analysis of metrics such as latency, response time, resource utilization, and power consumption. However, as a simulator, YAFS shares the inherent limitations in application execution and lacks support for the SDN paradigm.

Our proposed EFCC framework offers substantial differences compared to previously analyzed tools. EFCC enables the emulation of complete systems, including the applications to be tested, network elements (such as switches and SDN controllers), and terminal devices (hosts). Utilizing Docker containerization, all these components run real software, yielding results that are more reflective of actual conditions than those obtained with simulation tools. Applications can be divided into microservices following an MSA infrastructure, with some microservices capable of communicating with IoT devices through a Service Function Chaining (SFC) [11] workflow architecture. Additionally, EFCC focuses on the SDN network paradigm and can implement transparent service discovery within the SDN controller, allowing devices to communicate with microservices without needing to know their exact locations beforehand without significantly impacting performance metrics.

III. EFCC Framework

In this section, we detail the proposed EFCC framework by outlining its different modules. The scenario construction begins with a data input module, followed by a parser that builds the infrastructure. This setup allows for testing both network and application performance. EFCC consists of four main modules: (i) the DADOSIM scenario format (detailed in Section III-A), (ii) the parser algorithm (explained in Section III-B), (iii) the Kathará lab with its associated Docker containers (described in Section III-C), and (iv) the POX controller in an SDN network scenario, including service

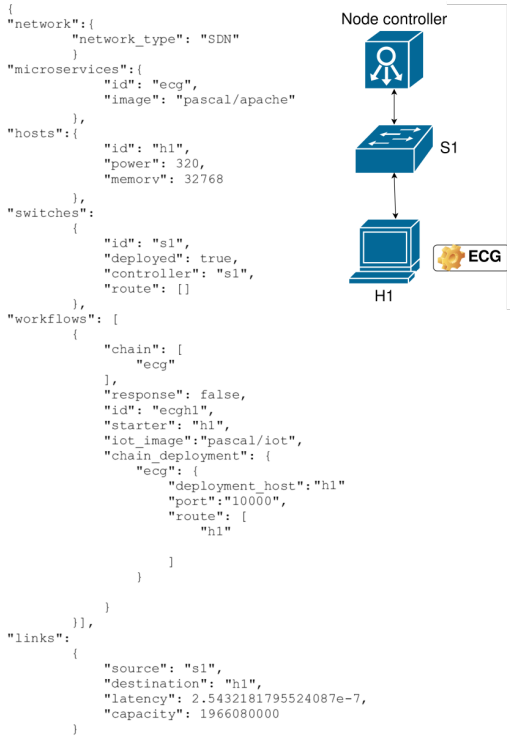


Fig. 1. Scenario example, including its description in the DADOSIM format.

discovery functionality and behavior in traditional IP networks (detailed in Sections III-D and III-E, respectively).

A. DADOSIM scenario format

Within EFCC, a *scenario file* expresses the complete configuration of an emulated testbed, including the deployment and configuration of the application, computing, and network layers. Scenario files have a JSON format that follows a specific schema named DADOSIM, as it is similar to the input format used by the DADO optimization framework [6]. An example EFCC scenario file, along with a visual representation of the scenario, is provided in Fig. 1. The DADOSIM format is structured in 6 blocks, which are described in the following:

- **Network**: This block defines the paradigm used by the network layer of the scenario. EFCC supports both traditional IP networks and SDN ones.
- **Microservices**: This block contains the list of the different microservices defined in the application's MSA. Each microservice must have its own unique identifier and define the Docker image that implements the microservice. This can refer to images in the local Docker image registry, or remote images that will be pulled when the emulation is started.
- **Hosts**: This block defines the computing layer of the scenario, i.e., its computing devices or hosts. Each must have a unique identifier, and its specifications in terms of power (in CPU percentage) and RAM (in bits).
- **Switches**: This block defines the SDN switches or IP switches and routers that integrate the networking fabric. A unique identifier also defines each switch. Moreover, the *deployed* attribute defines whether an SDN controller is or is not deployed to the switch, while the *controller* attribute defines the switch co-located with the SDN controller that the switch will communicate with. If the switch needs to communicate with another controller, the route used for its traffic can be defined in the *route* attribute. If no route is defined, the shortest path routing algorithm will be used, leveraging latency length metric.
- **Workflows**: This block defines the different workflows requested in the scenario. Each workflow must have a unique identifier. Moreover, they contain information about the IoT device requesting the workflow (*starter*), all the microservices they request (*chain*), and whether the data provided as output by the final microservice should be routed to the requesting IoT device (*response*). The workflow also defines the Docker image used by the requesting IoT device (*iot_image*), and details the deployment of the microservices (*chain_deployment*). For each requested microservice, the workflow defines the ID of the host in which it is deployed (*deployment_host*) and its port. Moreover, the path used to access this microservice from the previous one can be specified.
- **Links**: This final block defines the links that will connect the hosts and switches. Links do not need to have a unique identifier and are bidirectional, and their latency (in seconds) and capacity (in bytes) must be specified.

B. EFCC parser

EFCC automatically generates the container definitions, the underlying network links, and their configuration to automate the creation of a testbed that accurately represents a Computing Continuum scenario with the desired MSA-based IoT application running over an IP or SDN network. The remainder of this subsection details how the parser creates and configures containers, workflows, and network links.

1) Container creation: EFCC generates containers for some elements described in the DADOSIM files: switches, hosts, and microservices, although they are handled differently depending on their type. Beginning with switches, for each of the defined switches, a container is generated, which executes Quagga if the network is IP-based, and OpenVSwitch if the network is SDN-based. For example, switches *S1*, *S2*, *S3*, and *S4* from Fig. 2 become a separate containers in Fig. 3. EFCC automatically generates the necessary configurations in both cases (e.g. routing protocol daemons, bridge generation, execution mode) so the developer or operator does not need to manually do so. Moreover, for SDN networks, the SDN controller is also provided as a container, executing the custom POX EFCC controller described in Section III-D. Among the available SDN controllers, POX was selected over more recent alternatives due to its ease of understanding, reimplementation, and modification, to facilitate the reuse of the proposed code.

Hosts and microservices are handled differently: each microservice in MSA-based applications is packaged as a separate Docker image [2], and should be instantiated as a separate container. Hence, a host running multiple microservices cannot

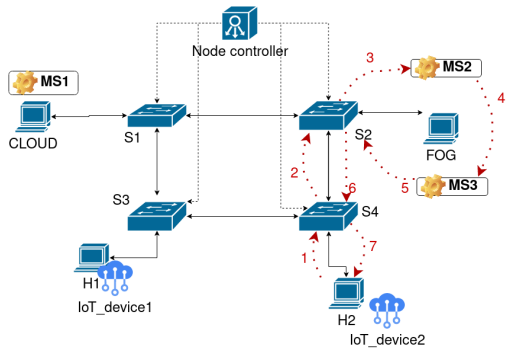


Fig. 2. Example scenario with 4 SDN switches, an SDN controller, 4 hosts, 3 microservices, 2 IoT devices, and 1 workflow.

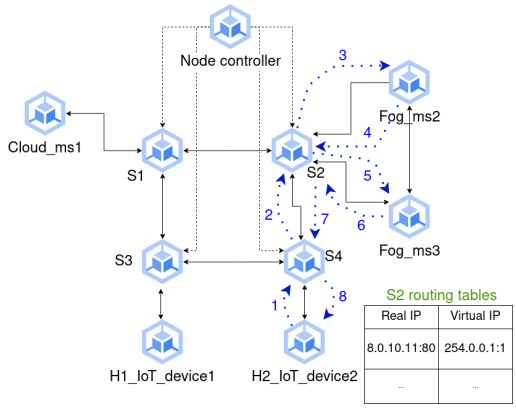


Fig. 3. Containers generated by EFCC from the scenario depicted in Fig. 2.

be directly converted to a single container. Instead, a container is generated for every microservice executed by a host. For example, a host running a single microservice, like the *Cloud* node in Fig. 2, becomes a single container (*Cloud_ms1*) in Fig. 3, while a host running two microservices (*Fog* in Fig. 2) becomes two containers (*Fog_ms2* and *Fog_ms3* in Fig. 3). Nonetheless, the hardware considerations (i.e. CPU, RAM) are still applied.

The networking across the generated containers is adjusted to ensure different containers that represent different microservices executed by the same host can communicate as if they were executed in the same machine. Furthermore, the general configurations for these containers (e.g. IP addresses, routing gateways, etc.) are automatically configured as well. Devices that request workflows, such as IoT devices, have another container created for them, using the image specified as the IoT image for the workflow in the scenario file, and are also automatically configured. This is shown as hosts *H1* and *H2* from Fig. 2 becoming a container each in Fig. 3.

2) Workflows: Workflows are the model with which IoT devices request application functionalities involving one or more microservices in order. The goal is to fulfil the request initiated by the IoT device, designated as the source node. In Fig. 2, a representation of a workflow is shown in red color. This workflow is requested by the IoT device of host *H2* (*H2_IoT_device2*) and requires microservices *MS2* and *MS3*

Algorithm 1 Pseudocode of the link creation subroutine

```
1: for all links ∈ scenario do
2:     if link.source is switch and link.destination is switch then
3:         domain ← createDomain(link.endA.ID + link.endB.ID)
4:         link.endA.registerDomain(domain)
5:         link.endB.registerDomain(domain)
6:     else
7:         host ← link.getHost()
8:         if not domain then
9:             domain ← createDomain(link.getS().ID + host.ID)
10:        end if
11:        link.getS().registerDomain(domain)
12:        if (host.countIoT() + host.countMicroservices()) > 1 then
13:            if not innerDomain then
14:                innerDomain ← createDomain(host.ID)
15:            end if
16:            for all containers ∈ host do
17:                container.registerDomain(innerDomain)
18:                container.registerDomain(domain)
19:                if not container.defaultGw then
20:                    container.defaultGw ← link.getS().IP(domain)
21:                end if
22:            end for
23:        else
24:            host.container.registerDomain(domain)
25:            if not host.container.defaultGw then
26:                host.container.defaultGw ← link.getS().IP(domain)
27:            end if
28:        end if
29:    end if
30: end for
```

located in the *Fog* node.

In the scenario file, users can specify the port where each microservice will be deployed on the host and the path the request will take to reach the host. This routing information is stored in a separate JSON file, which serves as the input for configuring the custom SDN controller. If a microservice lacks a predefined route, the shortest path between the source and destination hosts is determined.

3) Networking links and domains: The next part of the parser creates the links across the containers created in the previous parts. As some hosts are represented by multiple containers, EFCC generates not only the links stated in the scenario file, but also *virtual links* to connect different containers that represent the same host. Moreover, some individual links defined in the scenario file can become multiple links: the link between *S2* and *Fog* in Fig. 2 becomes two links, *S2-Fog_ms2* and *S2-Fog_ms3*, in Fig. 3. As EFCC leverages the Kathará [12] network emulator to create the testbed, links are represented by Kathará *domains*. A domain can be seen as the representation of a network cable: all network interfaces connected to the same domain are considered to be directly connected to each other.

As link creation is a complex process, the pseudocode for the link creation subroutine in EFCC is described by Alg. 1. This subroutine allows achieving network connectivity by creating and associating different domains with the containers that will act as switches, IoT devices, and microservices. First, if the link connects two switches, a domain is created and associated with both (lines 2-5). On the other hand, if the link connects a switch to a host (lines 6-23), different considerations are taken. First, a domain for the link is created using the switch and host's IDs (lines 8-10), and associated with the host (line 11). Then, it is necessary to know if the host has multiple containers to adapt the link generation (lines 12-23). If so, a virtual link is created through a virtual domain

(lines 13-15), and all containers are associated with the host are connected to both the virtual domain and the link's domain (lines 16-18). Moreover, if the containers had no gateway, the switch is defined as their gateway (lines 19-21). On the other hand, if the host has only one container, it is just connected to the link domain (line 24), and the switch serves as its gateway if no prior one existed (lines 25-27). This ensures all containers belonging to the same host are connected to the same virtual domain, separate from the rest of the network, as well as link multiplexing for switches connected to hosts that become multiple containers.

Each domain is configured as an independent IP subnet with automatic addressing. When a host joins a domain, EFCC creates the domain interface and assigns an IP address, allowing containers to be multi-homed (one interface per connected domain) and supporting scalable deployments. For example, in Fig. 3, *S2* is connected to 5 domains (*S2*-controller, *S2-S1*, *S2-S4*, *S2-Fog_ms2*, and *S2-Fog_ms3*), and thus has 5 different IP addresses, each belonging to a different domain.

C. Kathará laboratory and Docker containers

Kathará is a network emulation tool that allows to create and configure an emulated network topology that interconnects containers leveraging Docker technology [12], widely used for realistic network simulations [13], [14]. EFCC automates the process of building an emulated network testbed on top of Kathará by automatically configuring the network domains, interfaces, and containers. To do so, the EFCC parser generates a Kathará laboratory: a configuration of Kathará that allows replicable executions of a network environment, including the commands to be executed in each container, the network connectivity, the computing constraints of each container, and other internal configurations.

To create and configure containers for the topology, the parser generates a configuration file for each host named *hostname.startup*. This file is used by Kathará, enabling EFCC to specify the actions to be executed when each container starts. The configuration varies depending on the type of node being considered (IoT devices, microservices, or switches).

- **IoT devices and microservices**: IoT devices and microservices containers will configure their network interfaces using the `ifconfig` command, specifying the IP address, netmask, and broadcast addresses for each interface. The Traffic Control Linux tool (`tc` command) will be employed to limit the bandwidth and delay of the respective interfaces. Lastly, the `route` command will be used to set the default gateway for the host.
- **Switches**: The configuration file for a switch is more complex, as it includes a series of OpenVSwitch configuration commands. These commands connect the switch's interfaces with the necessary virtual ports. Network interfaces will be assigned their IP addresses using the `ifconfig` command, while their delay and bandwidth will be managed using the Traffic Control tool (`tc`).

EFCC then generates a Kathará laboratory configuration (*lab.conf*) file, which defines the configuration of the network topology to be created and emulated. After starting the scenario in Kathará, a Python script will be executed to verify that all containers have been correctly deployed. If any errors are detected, the script will restart the environment.

D. POX controller

The EFCC framework can deploy SDN-based testbeds, and therefore, a controller for the SDN network is required. EFCC utilizes a customized version of the POX controller to serve as the SDN controller due to its flexibility, extendability, and adaptability for different use cases. Additionally, this customized SDN controller implements a service discovery mechanism using workflows, enabling IoT devices to communicate with microservices without needing to know their specific locations.

1) Basic controller behavior: The following outlines the network functions required to ensure connectivity between IoT devices and microservices. The controller is responsible for installing the necessary flow rules in the SDN switches, enabling requests from IoT devices to reach microservices and allowing microservices to communicate with each other.

Communication between switches and the controller is facilitated by the OpenFlow protocol, which allows traffic flows to be managed within the SDN network. Routing decisions can be customized by the framework user, as described in Section III-A. If no routing is specified, the shortest path algorithm is applied instead. The EFCC parser will write all routing information for each path (IPs and ports) to an external file named *data.json*, which the POX controller can interpret.

2) Service discovery and virtual traffic: EFCC distinguishes between virtual traffic and real traffic. Real traffic includes traditional network traffic directed towards a known destination. Virtual traffic, on the other hand, requires service discovery since the destination address (including IP address and port) is not known in advance. This type of traffic typically involves requests for services from IoT devices, where the destination is a specific microservice within a particular workflow. Virtual traffic must use the TCP protocol , and target a virtual IP address and destination port. The POX controller then automatically handles service discovery, routing the traffic to the container hosting the required microservice for the specified workflow. This involves dynamic translation between virtual and real IP addresses and ports.

The destination port corresponds to the specific microservice being requested (e.g., port 1 refers to the first microservice). However, for the final response from the last microservice back to the IoT device, the non-reserved port 6000 is used. Microservices can receive requests from various IoT devices, with each request having a unique IP address tied to its specific workflow.

Requests within a workflow travel through the network following the routes specified in the scenario file. These routes are customized by the user in the route field within the workflows. If no routes are specified, the shortest path algorithm is used to determine them. To handle the routing of each request, the switches maintain both a virtual table and a real table (see Fig. 3) for switching different types of traffic.

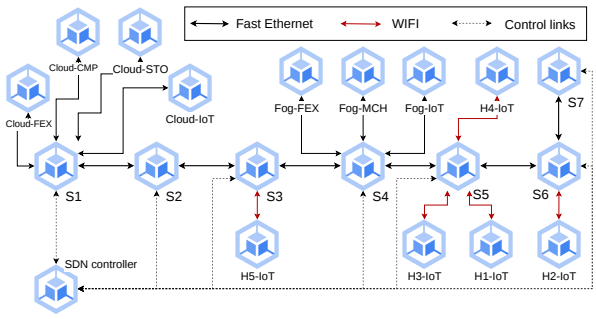


Fig. 4. Scenario used for testing.

The real routing tables are calculated by the EFCC parser using the shortest path algorithm, taking into account the destination IP address and outgoing port. The virtual routing tables are generated based on the construction of virtual requests and the customized routes each must follow. Virtual packets will follow the routes stored in the controller's virtual tables and will be directed through the final switch before reaching the host where the microservice is deployed. The controller translates the virtual packet's address to a real address, so the microservice only processes the real traffic.

Fig. 3 illustrates a sample topology featuring Docker containers and a workflow representation (highlighted in blue with various steps). In this scenario, host *H2* needs to request the microservice *MS2*, deployed on the *Fog* node, but it does not know the microservice's real IP address. The IoT device's request uses a virtual IP address and port: 254.0.0.1, port 1. The controller will automatically redirect the request according to the custom route (or the shortest path), which will be integrated into the virtual tables of each SDN switch (step 2). Upon reaching the final switch *S2* in the path, the controller installs a rule in its flow table to translate the packet's destination IP from a virtual address (e.g., 254.0.0.1, port 1) to a real address (e.g., 8.0.10.11, port 80). After this translation, the virtual IP becomes the real destination IP of the host where the *MS2* microservice is deployed. Consequently, the *MS2* microservice is unaware that the request is ultimately intended for the *MS3* microservice. In step 4, the virtual request travels to the first network node and, upon reaching the final node, the necessary translation is performed (prior to step 5). This workflow requires a response from the *MS3* microservice back to the IoT device, involving virtual request handling in steps 6, 7, and 8. The translation for this response is done at node *S4*.

3) POX component implementation: The custom POX component is implemented via a Python script used to build the controller's Docker image. This image is then assigned to the controller container in the *lab.conf* file. The controller container will run the POX component alongside the network data generated by the EFCC parser. The necessary data for routing and ensuring network connectivity is stored in the *data.json* file within the controller directory. This file contains detailed information on network switches, IP addresses for each container within a host, gateways for each container, user-defined routes, and the routing tables for all switches.

TABLE I
WORKFLOWS TESTED IN THE SCENARIO.

ID	Starter	First microservice	Second microservice	Third microservice
Cloud requests				
1	H1 - IoT	FEX	CMP	STO
3	H3 - IoT	FEX	CMP	STO
5	H5 - IoT	FEX	CMP	STO
7	Fog - IoT	FEX	CMP	STO
9	H2 - IoT	FEX	CMP	STO
11	H4 - IoT	FEX	CMP	STO
13	H1 - IoT	FEX	CMP	STO
15	H3 - IoT	FEX	CMP	STO
Fog requests				
ID	Starter	First microservice	Second microservice	Third microservice
2	H2 - IoT	FEX	MCH	-
4	H4 - IoT	FEX	MCH	-
6	Cloud - IoT	FEX	MCH	-
8	H1 - IoT	FEX	MCH	-
10	H3 - IoT	FEX	MCH	-
12	H5 - IoT	FEX	MCH	-
14	H2 - IoT	FEX	MCH	-

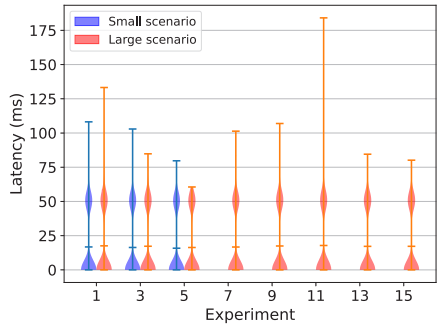
E. IP network support

Although the proposed framework focuses on SDN-based emulated testbeds, it is possible to perform the deployment over an IP network. In order to do this, it must be specified in the *network* field of the scenario file. Traditional IP and SDN paradigms have a number of differences that affect the behaviour of EFCC. IP networks do not leverage controllers, so it is not possible to implement the service discovery functionality. Instead, distributed routing (RIP, OSPF) is considered, implemented using Quagga, and automatically configured through the parser. The code for EFCC, is available as a Git repository [1].

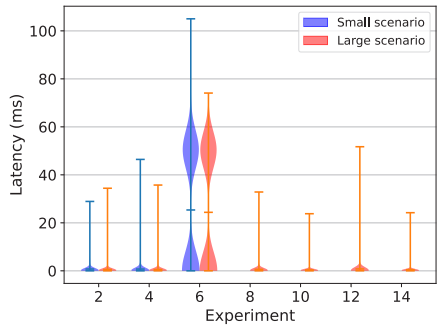
IV. EXPERIMENTAL EVALUATION

This section shows the functioning of EFCC over a realistic smart city use case based on the application in [15]. The application performs automated facial recognition to control facility access, registering authorized users and distinguishing them from intruders. To do so, the application comprises a total of four different microservice types. The first one, noted as *FEX* (*Feature EXtraction*), takes as input the video gathered by an IoT device, leveraging computer vision models to detect faces in the video, segment them, and extract the relevant features for facial recognition, outputting them [15]. The next microservice, labeled *MCH* (*MatCHing*), takes the extracted features as input, and compares them to the features of known users, outputting a boolean that grants or denies access to the facility [15]. These features are saved using the third microservice, *STO* (*STOrage*) [15]. However, due to the high dimensionality of the features, it is necessary to first compress them, a functionality provided by the *CMP* (*CoMPression*) microservice [15]. Therefore, two types of workflows exist, one for registering new users that should be granted access (FEX-CMP-STO), and one for checking whether a user can access the facility (FEX-MCH). This MSA is therefore deployed, with the aim of evaluating network performance and service discovery functionality in EFCC. The considered topology is represented in Fig. 4, which is composed of 7 SDN switches, 7 hosts, and an SDN controller.

[1]https://bitbucket.org/spilab/efcc/src/master/

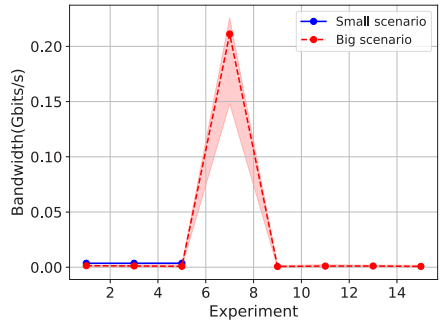


(a) Cloud requests

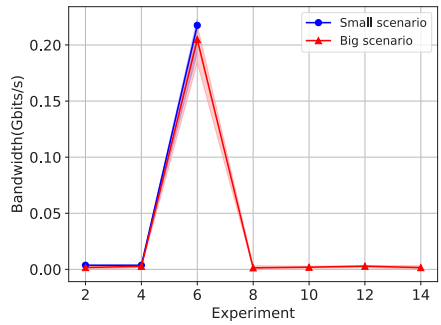


(b) Fog requests

Fig. 5. Average latency per experiment



(a) Cloud requests



(b) Fog requests

Fig. 6. Average bandwidth per experiment

A. *Performance experiments*

For this scenario, we examined two different scenario files. The first file represents a low-load scenario with only 6 workflows. In this setup, requests are sent from all IoT devices deployed across various hosts to the *Fog* and *Cloud* nodes. The fog layer, based on the OpenFog Reference Architecture (OFRA), serves as a processing layer close to the IoT devices within the network. Conversely, the cloud layer functions as a remote processing layer managed by an external provider. Therefore, it has been included in a separate test, as detailed in Table I. The second scenario file expands the number of workflows to assess scalability and its impact on QoS. It includes a total of 15 workflows requested by IoT devices. In addition to the 6 workflows from the first scenario, 9 additional workflows are introduced. These new workflows involve all IoT devices making requests to every microservice deployed across both *Fog* and *Cloud* nodes.

Table I shows the information of each of the IoT requests, including their ID, the starter node, and the workflow (i.e., chain of microservices). A differentiation of *Cloud* requests and *Fog* requests has been made due to their difference in the number of microservices per workflow. For each of the experiments, ten executions have been carried out and the average of the results obtained for both latency and bandwidth metrics was calculated.

After running the EFCC parser with the two scenario files, the results are presented in Figs. 5 and 6, which display the average latency and bandwidth per IoT request. Fig. 5a illustrates the results from workflows arriving at the cloud node, with the Ping tool used for this experiment. It is noted that for the first three experiments, which are common to both scenarios, the smaller scenario consistently exhibits lower latency (on average) compared to the larger scenario due to minor network congestion, specifically between 0.5 and 0.8 seconds less. Additionally, this figure shows higher latency values than those in Fig. 5b, which represents latency for requests made to the *Fog* node. This discrepancy arises because the *Cloud* node has a higher latency in its network access link (refer to Fig. 4), affecting the latency of all requests received.

Fig. 5b illustrates the latency of requests reaching the microservices deployed in the *Fog*. On average, results generally range between 0 ms and 1 ms, with the exception of experiment 6, which shows an outlier. In this case, the IoT device making the request is located within the *Fog* node. The difference in this experiment is due to the IoT device being connected via a Fast Ethernet link, which offers higher capacity and lower latency compared to the Wi-Fi links used by the other hosts (see Fig. 4).

After analyzing latency, Fig. 6a presents the results from the experiment measuring the bandwidth of requests arriving at the *Cloud* node. This measurement was conducted using the `iperf` tool. The results for the first three workflows indicate that bandwidth is higher in the smaller scenario with lower traffic volume. Similarly to Fig. 5b, experiment 7 shows a request from the Fog node to the *Cloud* node, where the access link is Fast Ethernet, resulting in higher bandwidth.

The final result, shown in Fig. 6b, presents the bandwidth measurements for the *Fog* node. These results are comparable to those obtained for the *Cloud* node (refer to Fig. 6a).

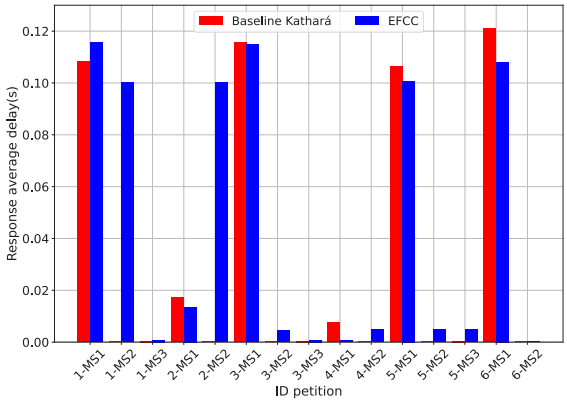


Fig. 7. Average response time in the service discovery experiments.

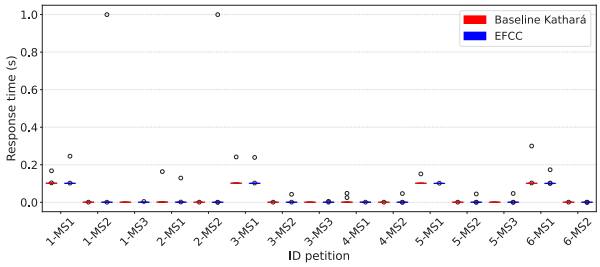


Fig. 8. Box plot representing the response times for the workflows in the service discovery experiments.

TABLE II
IMPACT OF SCENARIO SCALE ON MEMORY USAGE AND DEPLOYMENT TIME

Nodes	Workflows	Memory usage (MB)	Deployment time (s)
14	6	241	10
14	16	252	11
50	35	431	16
100	70	858	30

The lower-loaded scenario generally exhibits higher bandwidth than the larger one. Notably, experiment 6 displays an outlier due to the *Cloud* node sending requests to the *Fog* node via a Fast Ethernet link, which affects the bandwidth measurements.

B. Service discovery experiments

The service discovery implementation was tested using two scenarios: i) the same as before, using EFCC (Kathará + custom POX) with its virtual routing that performs service discovery (virtual scenario), and ii) a baseline scenario that combines Kathará with Faucet (real scenario). This analysis aims to evaluate the performance of virtual routing for service discovery and compare it with traditional traffic in terms of response latencies for various packets. The `tcping` tool was used for these tests, as service discovery operates at the transport layer. For each request, 10 samples were collected to derive statistical metrics. Each test corresponds to a request within a workflow, and the service discovery tests were labelled according to the workflow identifier and the microservices of the first 6 workflows listed in Table I.

Fig. 7 displays a bar chart illustrating the average time taken for each request to reach the target microservice and receive a response. The blue bars indicate the time required in the EFCC scenario, while the red bars represent the results from the baseline scenario. In the baseline scenario, microservices belonging to the same workflow are hosted on the same machine (*Cloud* or *Fog*). Consequently, the first request traverses the entire network. The chart reveals that the highest latencies are observed in the first microservice of each workflow. In two instances (workflow *2-MS2* and workflow *4-MS2*), this pattern does not hold, where the average time for service discovery performed on the host itself exceeds the average service discovery time from the IoT device on the same target node (both cases being *Fog*). This anomaly is likely due to an outlier among the 10 samples, which skews the mean result. These outliers are illustrated in Fig. 8, which features a box plot with very small boxes, indicating low variability in the results. Additionally, there is a significant gap with the outliers. Overall, the medians for the baseline (red line) and EFCC (blue line) tests for the same request are nearly identical (as shown on the graph), demonstrating that the service discovery process has been implemented with minimal performance degradation compared to traditional traffic.

C. Scalability tests of the deployment process.

In this section, we analyze both the computational cost in terms of time associated with emulating scenarios of different sizes using Kathará and the memory consumption of the emulation after deployment, thereby evaluating the scalability of the proposed framework. To this end, experiments are conducted on four scenarios: the two presented in the previous sections, consisting of 14 nodes with 6 and 16 workflows, respectively, and two additional scenarios. The first includes 50 nodes and 35 workflows, while the second comprises 100 nodes and 70 workflows.

Table II shows the results of completing the emulation for each scenario. As observed, both memory usage and deployment time increase with the scale of the scenario. However, the results suggest that the number of workflows does not significantly impact these metrics. Instead, the number of nodes appears to be the dominant factor, strongly influencing both memory consumption and deployment time. Overall, the growth remains moderate, indicating that the proposed framework is able to efficiently handle larger network sizes and workflow demands, thus demonstrating good scalability.

V. CONCLUSIONS

In this work, we propose the EFCC framework to assist researchers in evaluating their deployment models within SDN-Fog environments, where IoT applications following the MSA paradigm are implemented. Additionally, a service discovery mechanism was developed at the network level to automate the routing of requests from IoT devices. As there are no existing tools that integrate the three-layered approach (covering application, computing, and network dimensions), EFCC aims to support researchers and network operators in making informed decisions regarding network, computing, and application deployments while ensuring or optimizing QoS.

Furthermore, no known service discovery tools exist for SDN scenarios. Evaluation in a realistic smart city scenario demonstrates that EFCC can be extended to various scenarios and deployments needed by the Computing Continuum research community and for service discovery purposes.

ACKNOWLEDGEMENTS

This work has been co-financed 85% by the European Union, the European Regional Development Fund and the Regional Government of Extremadura. Managing Authority: Ministry of Finance. Project File Number: IB24102. Grant File Number: GR24099. Projects PID2024-155693NB-C41, 0289_SER65_PLUS_6_P and grant PTQ2024-013825 funded by MICIU/AEI/10.13039/50100011033. This work was supported by the Valhondo Calaff Institution.

REFERENCES

[1] "Cisco Annual Internet Report - Cisco Annual Internet Report (2018–2023) White Paper." [Online]. Available: https://shorturl.at/Xupns

[2] R. Pradhan and A. K. Dash, "An overview of microservices," in *ICDSMLA 2019*, A. Kumar, M. Paprzycki, and V. K. Gunjan, Eds. Singapore: Springer Singapore, 2020, pp. 620–625, DOI: 10.1007/978-981-15-1420-3_66.

[3] J. L. Herrera, J. Galán-Jimnez, J. Garcia-Alonso, J. Berrocal, and J. M. Murillo, "Joint optimization of response time and deployment cost in next-gen iot applications," *IEEE Internet of Things Journal*, vol. 10, no. 5, pp. 3968–3981, 2023, DOI: 10.1109/JIOT.2022.3165646.

[4] S. Dustdar, V. C. Pujol, and P. K. Donta, "On distributed com- puting continuum systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 4092–4105, 2023, DOI: 10.1109/TKDE.2022.3142856.

[5] J. L. Herrera, J. Galán-Jiménez, P. Bellavista, L. Foschini, J. Garcia-Alonso, J. M. Murillo, and J. Berrocal, "Optimal deployment of fog nodes, microservices and sdn controllers in time-sensitive iot scenarios," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6, DOI: 10.1109/GLOBECOM46510.2021.9685995.

[6] J. L. Herrera, J. Galán-Jiménez, J. Berrocal, and J. M. Murillo, "Optimizing the response time in sdn-fog environments for time-strict iot applications," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 17 172–17 185, 2021, DOI: 10.1109/JIOT.2021.3077992.

[7] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," *SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 473–478, sep 2012, DOI: 10.1145/2377677.2377767.

[8] V. Kochar and A. Sarkar, "Real time resource allocation on a dynamic two level symbiotic fog architecture," in *2016 Sixth International Symposium on Embedded Computing and System Design (ISED)*, 2016, pp. 49–55, DOI: 10.1109/ISED.2016.7977053.

[9] T. Goyal, A. Singh, and A. Agrawal, "Cloudsim: simulator for cloud computing infrastructure and modeling," *Procedia Engineering*, vol. 38, pp. 3566–3572, 2012, DOI: 10.1016/j.proeng.2012.06.412.

[10] I. Lera, C. Guerrero, and C. Juiz, "Yafs: A simulator for iot scenarios in fog computing," *IEEE Access*, vol. 7, pp. 91 745–91 758, 2019, DOI: 10.1109/ACCESS.2019.2927895.

[11] D. Bhamare, R. Jain, M. Samaka, and A. Erbad, "A survey on service function chaining," *Journal of Network and Computer Applications*, vol. 75, pp. 138–155, 2016, DOI: 10.1016/j.jnca.2016.09.001.

[12] G. Bonofiglio, V. Iovinella, G. Lospoto, and G. Di Battista, "Kathará: A container-based framework for implementing network function virtualization and software defined networks," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018, pp. 1–9, DOI: 10.1109/NOMS.2018.8406267.

[13] S. Laso, J. Berrocal, P. Fernández, A. Ruiz-Cortés, and J. M. Murillo, "Perses: A framework for the continuous evaluation of the qos of distributed mobile applications," *Pervasive and Mobile Computing*, vol. 84, p. 101 627, 2022, DOI: 10.1016/j.pmcj.2022.101627.

[14] A. García-Fernández, S. Laso, J. A. Parejo, J. Berrocal, A. Ruiz-Cortés, and J. M. Murillo, "Towards effective saas pricing design: A case study of ccsim," in *Service-Oriented Computing – ICSOC 2024 Workshops*, S. Kallel, C. Raibulet, I. Bouassida Rodriguez, N. Faci, A. Bennaceur, S. Cheikhrouhou, L. Ben Ayed, M. Sellami, E. Y. Nakagawa, and R. Ben Halima, Eds. Singapore: Springer Nature Singapore, 2026, pp. 157–168, DOI: 10.1007/978-981-96-7238-7_13.

[15] S. Wang, M. Zafer, and K. K. Leung, "Online placement of multi-component applications in edge computing environments," *IEEE Access*, vol. 5, pp. 2514–2533, 2017, DOI: 10.1109/ACCESS.2017.2665971.

José Gómez-delaHiz (Student Member, IEEE) is a Ph.D. student at the University of Extremadura, Spain, where he has been awarded with a Valhondo predoctoral fellowship. His current research focuses on UAV-based networking, Green networking, and Intent-Based Networking. Contact him at jagomezdh@unex.es

Juan Luis Herrera is a MSCA postdoctoral fellow in the Distributed Systems Group at TU Wien. He received a Ph.D. in Computer Science from the University of Extremadura in 2023. His main research interests include IoT, CC, service management, and sustainability.

Sergio Laso received the Ph.D. in Computer Science from the University of Extremadura (Spain) in 2023. He is currently working at the Department of Computer Systems and Telematics Engineering of the University of Extremadura (Spain) and as Industrial Ph.D at Gloin S.L. His research interests focus on the distributed systems and computing continuum architectures. Contact him at slasom@unex.es

Javier Berrocal (Member, IEEE) received the Ph.D. degree in Information Technologies for the University of Extremadura in 2014, where he is currently an associate professor. His main research interests are software architectures, pervasive systems, and computing continuum. Contact him at jberolm@unex.es

Jaime Galán Jiménez (Member, IEEE) received the Ph.D. in Information Technologies from the University of Extremadura in 2014, where he is currently an associate professor. His research interests are network digital twins, AI/ML for network management, and UAV-based networks. Contact him at jaime@unex.es.

Introducing Neural Mesh for Logical Synthesis Models

Péter Baranyi, and Ádám B. Csapo

Abstract—**The paper introduces the Neural Mesh, a novel architecture that serves as an alternative to conventional neural network models widely used in contemporary AI systems. In parallel, the paper introduces the term Logical Synthesis Model (LSM) as a complement to Large Language Models (LLMs). While LLMs primarily rely on statistical representations and language-based reasoning to acquire and generate knowledge, Neural Mesh-based LSMs focus on structured representations and engineering reasoning to represent, analyze, synthesize, and control systems. LLMs excel at learning and reasoning over human knowledge expressed through language, whereas LSMs aim to learn structured representations that support the "understanding" of physical systems. In this sense, LLMs and LSMs may play roles analogous to the cerebrum and the cerebellum in biological intelligence. The novelty of the paper lies in bridging and combining the concepts of TP models, TP model transformation, and neural-network architectures. The proposed Neural Mesh is mathematically equivalent to the well-established TP model function family. The TP model transformation provides a framework for constructing TP models with unique features that are particularly advantageous for systems and control applications. The contribution of the Neural Mesh is that it represents these unique features in the form of a special neural-network architecture, where these features are expressed as trainable neural connections and parameters. As a result, elements that are traditionally determined through an offline TP model transformation become directly tunable through learning, while preserving the mathematical structure and control-theoretic advantages of the TP model framework. Therefore, the Neural Mesh representation opens a future research direction in which the TP model transformation is effectively embedded into the training process itself. Rather than generating a TP model through a subsequent offline transformation, the corresponding TP-model components are directly constructed and tuned during system identification via neural-network training tools.**

I. INTRODUCTION

One of the fundamental limitations of neural-network-based system modelling – particularly in the case of dynamic system control – is that the resulting neural network obtained through system identification is generally not represented in a mathematical form that can be directly utilized by mainstream optimization and control-design methodologies. In dynamic system modelling and control, the mathematical representation of the model is of primary importance, as it determines whether established control theories can be interpreted and methods for controller synthesis, stability analysis, robustness verification, and performance optimization can be directly applied. Consequently, conventional neural networks remain largely black-box models in the context of system modelling and control, since the mathematical structures and control-theoretic properties embedded in the learned representation are generally not explicitly accessible for subsequent analysis, verification, and controller synthesis.

To address these limitations, the paper introduces the term Logical Synthesis Model (LSM), inspired by the notion of Large Language Models (LLMs). Logical Synthesis Models (LSMs) differ fundamentally from Large Language Models (LLMs). LLMs are primarily designed for knowledge representation and content generation within the domain of Generative AI, relying on statistical representations, language-based reasoning, and prompting techniques. In contrast, LSMs are intended for system representation within the domain of Engineering AI. Rather than modelling linguistic knowledge, LSMs learn structured representations of systems that support engineering reasoning, enabling subsequent analysis, synthesis, optimization, verification, and control design.

The LSM concept substantially constrains the structure of the Neural Mesh, as the architecture must be designed to represent the dynamics of physical systems as faithfully as possible. As a result, the Neural Mesh is naturally driven by the mathematical structure of physical systems and engineering models, rather than by the statistical language representations that characterize LLMs. Consequently, the Neural Mesh follows a considerably more structured architecture than conventional neural networks. The primary objective of training is not to discover the network structure itself, but rather to tune the parameters associated with a predefined mathematical representation implemented by the architecture. In this sense, learning in a Neural Mesh focuses predominantly on identifying the coefficients of the underlying mathematical model. This contrasts with many general-purpose neural network architectures, where both the network parameters and, in many cases, the network topology, layer composition, and structural configuration may also be subject to optimization.

An interesting observation is that the distinction between LLMs and LSMs exhibits a loose analogy to the relationship between the cerebrum and the cerebellum in biological intelligence. While the cerebrum is primarily associated with knowledge representation, abstraction, and high-level reasoning, the cerebellum operates in close interaction with sensory signals and the physical environment, contributing to the interpretation, coordination, and control of physical processes.

A further interesting parallel is that the cerebellum possesses

P. Baranyi (peter.baranyi@uni-corvinus.hu) and Á. B. Csapó are with the Corvinus Institute for Advanced Studies (CIAS) and Institute of Data Analytics and Information Systems at the Corvinus University of Budapest, Budapest, Hungary and the Hungarian Research Network (HUN-REN). The authors declare that there is no conflict of interest regarding the publication of this paper. The research work was supported by National Research, Development and Innovation Office HURIZONT PROJECT 2025- 2026 NKFIH - 2024-1.2.3-HURIZONT-2024-00030.

DOI: 10.36244/ICJ.2026.2.3

a remarkably regular and highly structured architecture. It is organized into a relatively small number of well-defined layers, and learning is believed to occur primarily through the adaptation of synaptic connections rather than through changes in the overall architectural organization. Similarly, the Neural Mesh adopts a largely predefined structure motivated by the requirements of system representation, while learning is primarily associated with the adjustment of the parameters embedded within this structure.

A. Novel contribution

The paper introduces the Neural Mesh, whose mathematical transfer function is identical to one of the most widely used polytopic system representations in modern control theory. More specifically, the Neural Mesh and higher-order Tensor Product (TP)-based polytopic models [1]–[10] share exactly the same mathematical formulations. Consequently, the extensive body of control design methodologies developed for TP model transformation, TP-based polytopic representations, polytopic models in general, and Takagi–Sugeno (TS) fuzzy model representations can be directly applied to Neural Mesh models.

This connection provides immediate access to a broad range of established analysis, verification, and controller synthesis techniques. In particular, the Parallel Distributed Compensation (PDC) framework [11] becomes directly applicable to Neural Mesh representations. Since PDC-based controller synthesis can be formulated through Linear Matrix Inequalities (LMIs) [12], one of the most extensively investigated optimization frameworks in modern control theory, the Neural Mesh naturally inherits a rich set of mathematically rigorous tools for stability analysis, performance optimization, robustness verification, and controller design.

A natural question arises: if the transfer function realized by the Neural Mesh is mathematically identical to the model obtained through Tensor Product (TP) model transformation, which itself is based on a Higher-Order Singular Value Decomposition (HOSVD) structure and various convex-hull representations, why is it worthwhile to introduce and study the same mathematical representation in the form of a Neural Mesh?

The answer lies in the fundamentally different perspectives of the TP model, TP model transformation and Neural Mesh. The TP model transformation determines which components of the TP model should be constructed and how they should be configured in order to obtain advantageous properties from the perspective of systems and control theory (this is the reason why the general TP model form is popular in system theories). The Neural Mesh, in turn, represents the TP model in a neural-network structure in which these components are reduced to standard neural connections and trainable parameters commonly used in neural networks. Consequently, the determination of TP model components is no longer performed solely through an offline transformation procedure but can be achieved through a tunable learning process.

In this sense, the Neural Mesh transforms TP model construction and optimization from an offline execution paradigm into a trainable neural-learning paradigm while preserving the underlying mathematical and control representation. This representation also makes the individual components responsible for key TP-model properties – such as convex-hull manipulation – which reduce to simple weighting-matrix operations, explicitly identifiable within the neural architecture. As a result, these components become directly accessible to training and optimization procedures developed for neural networks, while simultaneously retaining the theoretical advantages and interpretability offered by the TP model transformation framework.

Therefore, an important aspect is that the Neural Mesh formulation makes the role of convex-hull optimization during learning more explicit. This is significant because a recently (2026) established formal mathematical result shows that one of the most critical, if not the most critical, elements of polytopic PDC design is the determination of the optimal convex hull [10]. This observation challenges the common view in the literature that optimality can be considered primarily at the level of the controller design once a polytopic model has been fixed. The influence of convex hull manipulation in TP model forms has already been observed in some engineering control-design applications, where different convex hull structures often lead to significantly different design outcomes [13]–[32].

B. Opening future research directions

From the above perspectives, the objective of learning in Neural Mesh models is not limited to achieving modelling accuracy. Rather, learning may also be interpreted as the search for an optimal convex-hull representation that supports subsequent analysis, verification, and controller synthesis. In the Neural Mesh architecture, this convex-hull structure appears as an explicit and visible structural component, thereby opening a new research direction on how the model can be trained such that the learning process itself leads to convex-hull optimization.

Conversely, this interpretation also opens another future research direction: operations and transformations between Neural Mesh models may be formulated as learnable procedures corresponding to transformations and operations between TP-based polytopic models. In this way, the Neural Mesh provides not only a learnable representation of TP-type polytopic models, but also a possible learning-based framework for carrying out transformations between such representations published in [5], [7]–[9].

In summary, although the transfer function of the Neural Mesh is mathematically identical to the representation obtained through TP model transformation, the Neural Mesh offers a substantially different perspective on the same underlying mathematical structure. Its architecture exposes structural components that can be more naturally connected to learning and training methodologies developed for neural networks, while simultaneously making explicit those elements that are of particular importance from a control-theoretic viewpoint.

II. NOTATION

The following notation is used in the paper:

- Indices are denoted by $n, r, c, j, l \ldots$ with corresponding upper bounds denoted as $N, R, C, J, L \ldots$ respectively.
- Scalars, vectors, matrices and tensors (multidimensional matrices) are denoted as $b \in \mathbb{R}$, $\mathbf{b} \in \mathbb{R}^{J}$, $\mathbf{B} \in \mathbb{R}^{J_1 \times J_2}$, $\mathcal{B} \in \mathbb{R}^{J^O}$. E.g. notation $\mathcal{B} \in \mathbb{R}^{C^N \times J^O}$ is equivalent to $\mathcal{B} \in \mathbb{R}^{C_1 \times \ldots \times C_N \times J_1 \times \ldots \times J_O}$.
- $[\mathcal{B}]_{index}$ addresses elements, e.g. $[\mathcal{B}]_{c_1,c_2,\ldots c_N} = \mathbf{B}_{c_1,c_2,\ldots c_N} \in \mathbb{R}^{J^2}$ of $\mathcal{B} \in \mathbb{R}^{C^N \times J^2}$;
- $\Omega \subset \mathbb{R}^N$ denotes a domain defined as the Cartesian product of one-dimensional intervals $\omega_n = [\omega_n^{min}, \omega_n^{max}] \subset \mathbb{R}$, i.e., $\Omega = \omega_1 \times \ldots \times \omega_N$.
- $\bigboxtimes_{n=1}^{N}$ denotes the vector(matrix)-tensor product, that is defined as

$$\mathcal{B} \bigboxtimes_{n=1}^{N} \mathbf{f}_n(x_n) = \tag{1}$$

$$= \sum_{c_1=1}^{C_1} \sum_{c_2=1}^{C_2} \cdots \sum_{c_N=1}^{C_N} \prod_{n=1}^{N} [\mathbf{f}_n(x_n)]_{c_n} [\mathcal{B}]_{c_1,c_2,\ldots c_N}.$$

III. Neural Mesh Architecture

The Neural Mesh maps an input vector $\mathbf{x} = [x_1, \ldots, x_N] \in \Omega \subset \mathbb{R}^N$ through four stages: (1) a per-dimension *Resolution Layer* composed of triangular activation functions that define subregions of the input domains; (2) a per-dimension *Complexity Layer* that classifies the input values through meaningful weightings – similarly to fuzzy membership functions – while characterizing the rank of the Neural Mesh; (3) a *Mesh Layer* (or *Core Mesh*) that forms the mesh (tensor-product) combination of the per-dimension outputs of the Complexity Layer; and (4) an *Output Layer* that weights and aggregates the outputs of the Mesh layer.

A. Resolution Layer

The Resolution Layer, i.e. the input layer, consists of R neurons, where R defines the "Resolution" of the representation. As R increases, the sensitivity – or equivalently, the resolution – with respect to the input also increases, see Figure 1.

Each neuron has one input x and one output s_r. Each neuron senses one subregion of the input space. The activation function of a neuron defines the subregion of the input space in which that neuron is active. Therefore, each neuron has an offset input g_r, and to ensure the systematic overlapping between subregions of the neurons, neighbouring neurons share their offset information, thereby enforcing a consistent boundary between their respective activation domains.

Consequently, the combination of the neuron-specific offsets and activation functions as shown on the upper plot in Figure 1 – induces a partitioning of the input space into subregions. Each subregion is assigned to the neighbouring neurons as depicted in Figure 1.

The activation function – that is, the neuron output – is denoted by $s_r(x, g_r)$, where "s" refers to "smoothing". In the present case, piecewise linear activation functions are applied; however, bell-shaped or various higher-order functions can also be employed. Therefore the activation functions $s_r(x, g_r)$ contribute to the smoothness of the output generated by the entire Neural Mesh.

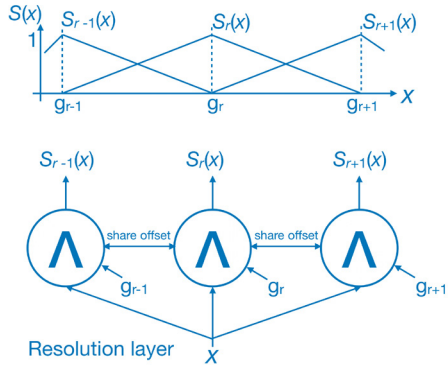


Fig. 1: Resolution Layer: Input Neuron and the activation functions

The output of the Resolution Layer is stored in a vector $\mathbf{s}(x, \mathbf{g}) \in \mathbb{R}^R$ with offset vector $\mathbf{g} = \begin{bmatrix} g_1 & \ldots & g_R \end{bmatrix} \in \mathbb{R}^R$, where $g_r < g_{r+1}$, as

$$\mathbf{s}(x, \mathbf{g}) = \begin{bmatrix} s_1(x, \mathbf{g}) & s_2(x, \mathbf{g}) & \ldots & s_R(x, \mathbf{g}) \end{bmatrix} \tag{2}$$

such that if $x \in [g_r, g_{r+1}]$ then

$$s_r(x, \mathbf{g}) = 1 - \frac{x - g_r}{g_{r+1} - g_r} \text{ and } s_{r+1}(x, \mathbf{g}) = 1 - s_r(x, \mathbf{g}), \tag{3}$$

and all other $s_z(x, \mathbf{g}) = 0$ ($z = 1, ..., R$ and $\neq r, r+1$), see the upper plot in Figure 1.

When the Neural Mesh has N inputs x_n, each input is associated with a Resolution Layer containing R_n neurons. The outputs of the Resolution Layers are:

$$\mathbf{s}_n(x, \mathbf{g}_n) = \begin{bmatrix} s_{n,1}(x_n, \mathbf{g}_n) & \ldots & s_{n,R_n}(x_n, \mathbf{g}_n) \end{bmatrix}, \tag{4}$$

where the offset is defined for each input by vector $\mathbf{g}_n = \begin{bmatrix} g_{n,1} & \ldots & g_{n,R_n} \end{bmatrix}$.

Remark 1: Note that at most two neighbouring neurons per input are active for any input x_n. Thus, the computation is reduced to these two neurons.

B. Output Layer

Each output y_l of the Output Layer is produced by a single additive neuron that has no activation function. Each neuron generates a weighted sum of the outputs of the previous layer, see Figure 2. Let the output of the previous layer be stored in the vector $\mathbf{p} \in \mathbb{R}^M$. Then, the activation at the Output Layer is given by

$$y_l = \mathbf{p}^T \mathbf{b}_l,$$

where the vector $\mathbf{b}_l = \begin{bmatrix} b_{l,1} & b_{l,2} & \ldots & b_{l,M} \end{bmatrix}^T \in \mathbb{R}^M$ contains the gains b_m of the connections between the previous layer and the neuron in the Output Layer, see Figure 2.

If the output has a higher-order structure, e.g. the output values are stored in a tensor structure $\mathcal{Y} \in \mathbb{R}^{J^O}$, that means that the linear index "l" is replaced with multi-linear indexing $j_1, j_2, \ldots j_O$ – the output is determined as

$$[\mathcal{Y}]_{j_1,j_2,\ldots j_O} = \mathbf{p}^T \mathbf{b}_{j_1,j_2,\ldots j_O}. \tag{5}$$

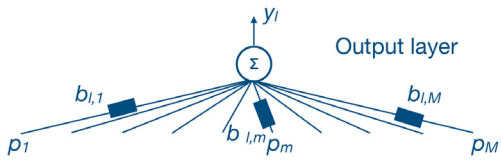


Fig. 2: Output Layer

This can be given by tensor-vector product as

$$\mathcal{Y} = \mathcal{B} \times_1 \mathbf{p}, \tag{6}$$

where the tensor of the gains is constructed as $\mathcal{B} \in \mathbb{R}^{M \times J^O}$, e.g. it contains vectors $\mathbf{b}_{j_1,j_2,\ldots j_O} \in \mathbb{R}^M$.

C. Complexity Layer

The Internal Complexity Layer is constructed by extending the Resolution Layer with an Output Layer, see Figure 3. It composes a higher-level information representation of the inputs. The Complexity Layer has C neurons and outputs. It characterizes how complex or simple the contribution of the underlying input is to the output of the Neural Mesh. Mathematically, in tensor algebraic terms, this corresponds to the rank of the input representation. Specifically, the number of neurons in the Complexity Layer can express the input rank of the Neural Mesh. If C is larger than the rank of the information associated with this input, then redundant neurons appear in the Neural Mesh. If C is smaller than the rank, the Neural Mesh cannot achieve the best possible approximation. Therefore, during training, C is increased up to at most the rank, with the objective of finding an appropriate balance between the number of neurons and the approximation accuracy of the Neural Mesh. This can also be associated with the micro or macro level of the input-output mapping of the Neural Mesh, or can be interpreted or readily linked with principal component analysis and singular functions. A further important point is that the functions implemented by the individual neurons can be trained in such a way that they represent the input information in a meaningful and interpretable form, similarly to fuzzy sets, see Section III/F.

The overall transfer function of the Resolution Layer and Complexity Layer is

$$f_c^C(x) = \sum_{r=1}^{R} s_r(x, \mathbf{g}) u_{r,c} = \mathbf{s}(x, \mathbf{g}) \mathbf{u}_c, \tag{7}$$

which leads to

$$\mathbf{f}^C(x) = \begin{bmatrix} f_1^C(x) & \ldots & f_C^C(x) \end{bmatrix} = \mathbf{s}(x, \mathbf{g}) \mathbf{U}, \tag{8}$$

where matrix $\mathbf{U} \in \mathbb{R}^{R \times C}$ contains the connection gains $u_{r,c}$.

When the Neural Mesh has N inputs x_n, each input is transferred through the Resolution Layer and the Complexity Layer as

$$\begin{aligned} \mathbf{f}_n^C(x_n) &= \begin{bmatrix} f_{n,1}^C & (x_n) & \ldots & f_{n,C_n}^C(x_n) \end{bmatrix} \\ &= \mathbf{s}_n(x_n, \mathbf{g}_n) \mathbf{U}_n. \end{aligned} \tag{9}$$

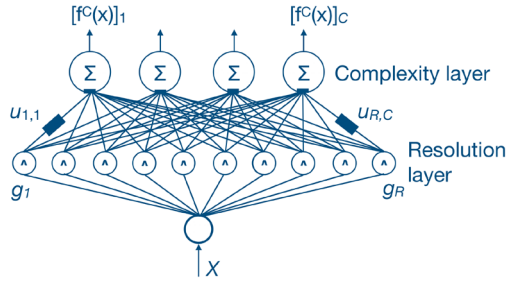


Fig. 3: Complexity Layer

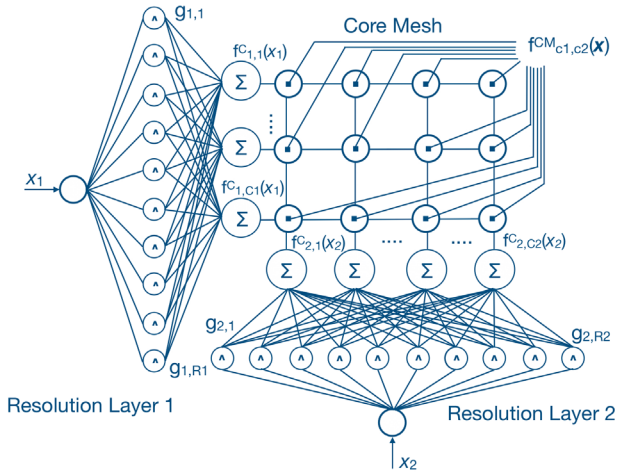


Fig. 4: Two-input Core Mesh

D. Core Mesh

This is not a layer in the classical sense; rather, it forms a hypercube-structured arrangement of multiplicative neurons (as opposed to additive neurons used above) with no activation function. Therefore, we also name this "layer" the Core Mesh. The inputs of the Core Mesh are the output vectors $\mathbf{f}_n^C(x_n)$ of the Complexity Layers of the inputs x_n. The tensor-valued output $\mathcal{F}^{CM}(\mathbf{x}) \in \mathbb{R}^{C^N}$ of the Core Mesh is generated by the tensor product of these inputs as:

$$\begin{aligned} [\mathcal{F}^{CM}(\mathbf{x})]_{c_1,c_2,\ldots c_N} &= \\ = \prod_{n=1}^{N} \left(\sum_{r=1}^{R_n} s_{n,r}(x_n, \mathbf{g}_n) u_{n,r,c_n} \right) &= \prod_{n=1}^{N} f_{n,c_n}^C(x_n). \end{aligned} \tag{10}$$

A two-input case is depicted in Figure 4.

E. Complete Neural Mesh Architecture

We arrive at the complete Neural Mesh architecture, when we add a final Output Layer having connection gains $b_{c_1,c_2,\ldots c_N}$ to the Core Mesh. Therefore each output of the Core Mesh is multiplied by $b_{c_1,c_2,\ldots c_N}$ as

$$[\mathcal{F}^{CM}(\mathbf{x})]_{c_1,c_2,\ldots c_N} b_{c_1,c_2,\ldots c_N} \tag{11}$$

and then summed as

$$f(\mathbf{x}) = \sum_{c_1=1}^{C_1} \ldots \sum_{c_n=1}^{C_N} [\mathcal{F}^{CM}(\mathbf{x})]_{c_1,c_2,\ldots c_N} [\mathcal{B}]_{c_1,c_2,\ldots c_N}. \tag{12}$$

When we substitute the above components we have

$$f(\mathbf{x}) = \sum_{c_1=1}^{C_1} \dots \sum_{c_n=1}^{C_N} \prod_{n=1}^{N} f_{n,c_n}^{C}(x_n)[\mathcal{B}]_{c_1,c_2,\dots c_N} = \quad (13)$$

that is

$$f(\mathbf{x}) = \quad (14)$$

$$\sum_{c_1=1}^{C_1} \dots \sum_{c_n=1}^{C_N} \prod_{n=1}^{N} \left(\sum_{r=1}^{R_n} s_{n,r}(x_n, \mathbf{g}_n) u_{n,r,c_n} \right) [\mathcal{B}]_{c_1,c_2,\dots c_N}$$

which can be written using the tensor-vector product as

$$f(\mathbf{x}) = \mathcal{B} \bar{\boxtimes}_{n=1}^{N} (\mathbf{s}_n(x_n, \mathbf{g}_n)\mathbf{U}_n) \,. \quad (15)$$

A 2D input case is depicted in Figure 5. A 3D input case is depicted in Figure 6.

Consider a case when $\mathbf{f}(\mathbf{x})$ is a vector-valued function. Then each output $f_l(\mathbf{x})$ has its own gains as

$$[\mathbf{f}(\mathbf{x})]_l = \sum_{c_1=1}^{C_1} \dots \sum_{c_n=1}^{C_N} \prod_{n=1}^{N} f_{n,c_n}^{C}(x_n)[\mathcal{B}_l]_{c_1,c_2,\dots c_N} \,. \quad (16)$$

This means we can create an $N+1$ dimensional tensor $\mathcal{B} \in \mathbb{R}^{C^N \times L}$ where the size of the last dimension is L, such that each entry $c_1, c_2, \dots c_N$ is a vector with size L. Therefore the above product becomes

$$\mathbf{f}(\mathbf{x}) = \mathcal{B} \bar{\boxtimes}_{n=1}^{N} (\mathbf{s}_n(x_n, \mathbf{g}_n)\mathbf{U}_n) \,. \quad (17)$$

Consider a case when we have a higher-order structure with a tensor-valued function $\mathcal{F}(\mathbf{x}) \in \mathbb{R}^{J^O}$. Then each entry $c_1, c_2, \dots c_N$ of $\mathcal{B}$ must be a tensor with the size of $J_1 \times \dots \times J_O$. This leads to

$$\mathcal{F}(\mathbf{x}) = \mathcal{B} \bar{\boxtimes}_{n=1}^{N} (\mathbf{s}_n(x_n, \mathbf{g}_n)\mathbf{U}_n) \,, \quad (18)$$

where $\mathcal{B} \in \mathbb{R}^{C^N \times J^O}$. Note that this transfer function of the Neural Mesh is identical to TP model form resulted by the TP model transformation [1]–[10].

IV. Design-Oriented Representation, and Formal Verifiability

The outputs of the Complexity Layer, referred to as the complexity functions, are equivalent to the weighting-function system of the TP model. These functions are also known as antecedent membership functions in fuzzy-system representations and scheduling functions in polytopic system representations. A recent publication [10] highlights their crucial role in control design and optimization.

The TP model transformation provides various tools for manipulating these functions in order to obtain favorable control-theoretic properties. In the Neural Mesh, however, the manipulation of the corresponding weighting functions is reduced to the adjustment of the connection-weight matrices $\mathbf{U}_n$. Consequently, structural modifications that would traditionally be performed through TP model transformation can be interpreted as tuning and training operations within the Neural Mesh architecture.

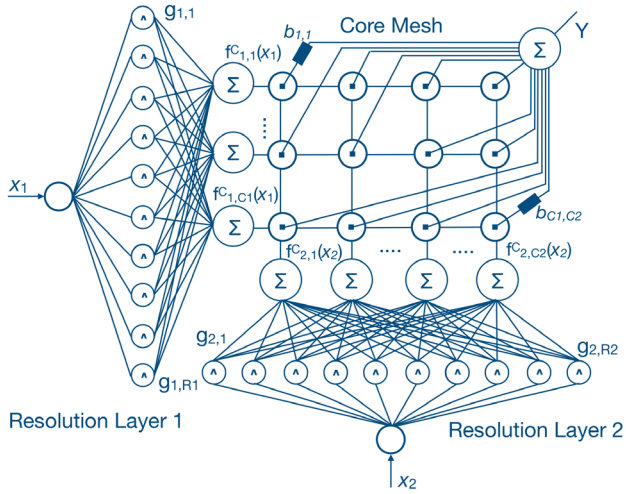


Fig. 5: Two-input Neural Mesh

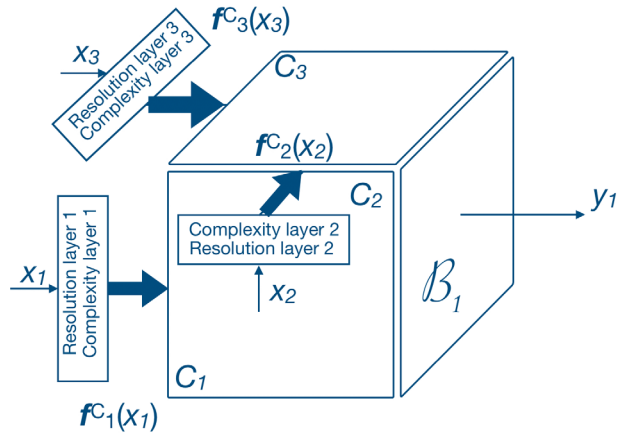


Fig. 6: Three-input, single-output Neural Mesh

Therefore, if further constraints are imposed on the outputs of the Complexity Layer, well-known polytopic formulations widely used in modelling and – for instance – control-theoretic research domains can be obtained. Specifically, if we enforce that the sum of the complexity functions $f_{n,c}^{C}(x_n)$ is always 1 on each dimension and are non-negative – a condition referred to as the Sum-Normalized and Non-Negative (SNNN) condition [1], [33] – as

$$\forall n, x_n : \sum_{c=1}^{C_n} f_{n,c}^{C}(x_n) = 1 \text{ and } \forall n, c, x_n : 0 \leq f_{n,c}^{C}(x_n) \quad (19)$$

then the resulting representation becomes convex, meaning that the output always remains within the convex hull spanned by the vertices $[\mathcal{B}]_{c_1,c_2,\dots c_N}$. Such convex formulations constitute fundamental requirements in several application domains, particularly in control theory and polytopic control design. This also contributes to the boundedness and numerical stability of the Neural Mesh output.

Furthermore, if the CNO (Close-to-Normalized) or IRNO (Inverse-Relaxed-Normalized) [1], [33] condition is imposed

on $\mathbf{f}_n^C(x_n)$, then these functions can naturally express meaningful classifications of the input domain, similarly to how membership functions in fuzzy logic provide a comprehensive representation of a given domain [1]–[3].

We note that the reason why the Neural Mesh uses triangular activation functions in the Resolution Layer, as described in Eq. (3), is that by enforcing these algebraic constraints such as SNNN, CNO via simple matrix operations on U, the model maintains a valid polytopic representation throughout training. This structural guarantee simplifies downstream evaluation steps, such as LMI feasibility checks, although computational scalability remains a consideration for high-dimensional inputs. Imposing the SNNN or CNO constraints on matrix $\mathbf{U}$ – resulting in simple matrix operations – is equivalent to imposing them on the Complexity functions at the output of the Complexity Layer, see Eq. (9). This follows from the closure property of SNNN and CNO matrices under matrix multiplication. Since $\mathbf{s}_n(x_n, \mathbf{g}_n)$ is SNNN or CNO, the product $\mathbf{s}_n(x_n, \mathbf{g}_n)\mathbf{U}_n$ remains SNNN or CNO, respectively, whenever $\mathbf{U}_n$ is chosen to be SNNN or CNO; see [1]–[3]. This has several advantages: (i) convexity is guaranteed by construction at every step of training, meaning the Neural Mesh always represents a valid polytopic model – this enables interleaving learning with control-design procedures such as LMI feasibility evaluation; (ii) the non-negativity and sum-to-one properties of the weights eliminate the vanishing-product problem in the Core Mesh (see Section V-D1); (iii) no nonlinear activation function is needed in the Complexity Layer, thereby preserving the pure HOSVD-based tensor-product structure; and (iv) the constrained rows of $\mathbf{U}$ are directly interpretable as importance distributions over the Complexity Layer cells.

V. Training of the Neural Mesh

All parameters of the Neural Mesh – namely the resolution parameters R_n, the subregions defined by $\mathbf{g}_n$, the weights between the Resolution Layer and Complexity Layer defined by $\mathbf{U}_n$, the number of neurons (rank) in the Complexity Layer defined by C_n, and the weights in the Output Layer defined by the tensor $\mathcal{B}$ – are subject to training. In practice, it is reasonable to start with lower values of the structural parameter R_n in order to capture the overall characteristics of the entire domain, and then to gradually increase R_n to achieve finer resolution. Similar considerations apply to C_n. First, a low-rank Neural Mesh is trained, which essentially captures the dominant or main components of the input-output mapping. Subsequently, the rank of the Neural Mesh can also be increased gradually by introducing additional neurons into the Complexity Layer.

A. Loss Functions

For regression tasks, the raw output $\mathbf{f}(\mathbf{x})$ is used directly and the model is trained with mean squared error (MSE).

For classification tasks with K classes, the output of the Neural Mesh produces K logits and a softmax is applied:

$$\hat{p}_k(\mathbf{x}) = \frac{\exp([\mathbf{f}(\mathbf{x})]_k)}{\sum_{k'=1}^{K} \exp([\mathbf{f}(\mathbf{x})]_{k'})}, \tag{20}$$

and the model is trained with the cross-entropy loss $\mathcal{L} = -\sum_k t_k \log \hat{p}_k$, where t_k is the one-hot target.

B. Optimization

The implementation supports Adam, AdamW, and SGD optimisers. In practice, Adam with a suitable learning rate and mini-batches provides fast convergence. Optional weight decay can be applied for regularization. All parameters of the Neural Mesh – the membership centers (and optionally supports), Complexity Layer weights, and Output Layer weights – are differentiable and trained end-to-end via backpropagation.

C. Setting the Resolution and Complexity

In many cases, the Resolution parameters R_n and the subregions $\mathbf{g}_n$ can be specified independently of the training procedure according to the sensitivity of the individual inputs. Similarly, the number of neurons in the Complexity Layer, defined by C_n, can also be adjusted based on different design considerations.

Increasing the Resolution primarily improves the granularity of the approximated input–output mapping. In contrast, increasing the number of neurons in the Complexity Layer becomes beneficial when further increasing the Resolution no longer improves the approximation performance. In such cases, the representational rank of the Neural Mesh must be increased by introducing additional neurons into the Complexity Layer.

The insertion of additional neurons into the Resolution Layer can be performed in a straightforward manner. Specifically, when a new neuron is inserted between two neighbouring neurons, a new row is inserted into the corresponding matrix $\mathbf{U}_n$ between the rows associated with the neighbouring neurons. The elements of the newly inserted row are determined through interpolation between the two neighbouring rows according to the relative distances of the new neuron from its neighbours.

Consider the case where an additional neuron with offset $g_{n,a}$ is inserted on input n between two neighbouring neurons with offsets $g_{n,r}$ and $g_{n,r+1}$, such that $g_{n,r} < g_{n,a} < g_{n,r+1}$. Then, the weights corresponding to this additional neuron are obtained by linear interpolation between the weights of the two neighbouring neurons.

$$[\mathbf{U}_n]_a = (1-\lambda)[\mathbf{U}_n]_r + \lambda[\mathbf{U}_n]_{r+1}, \tag{21}$$

where

$$\lambda = \frac{g_{n,a} - g_{n,r}}{g_{n,r+1} - g_{n,r}}. \tag{22}$$

This operation does not change the input–output mapping of the Neural Mesh; hence, training can proceed without having to restart from the beginning.

In a similar manner, inserting an additional neuron into the Complexity Layer corresponding to input n is also straightforward. Consider the case where the number of neurons in the Complexity Layer associated with input n is increased from C_n to C_n+1. Then, the tensor $\mathcal{B}$ must be extended with zeros along the corresponding dimension. This operation also does

not change the previously learned input–output mapping of the Neural Mesh.

Once $\mathbf{g}_n$ and C_n are fixed, standard gradient-based training procedures can be applied to further optimize $\mathbf{U}_n$ and $\mathcal{B}$.

D. Further Considerations for Training

Although the network structure and training procedure outlined above is already well-suited to many problem domains, the effectiveness of the training procedure can be further improved using the following methods.

1) Resolution Weight Normalization: The Core Mesh computes the product of per-dimension Complexity Layer activations; when any single factor is zero, the entire product – and its gradient with respect to all other factors – vanishes.

In such cases, it can be helpful to address this "vanishing product" problem by normalizing the weight matrix $\mathbf{U}_n$ prior to the linear transformation in the Complexity Layer. The normalization operates column-wise (per Resolution Layer cell r, across all C_n Complexity Layer cells) and ensures non-negative weights that sum to one. Three modes are available:

- **None**: raw weights are used as-is.
- **Softmax**: $\hat{u}_{c,r} = \exp(u_{c,r}) / \sum_{c'} \exp(u_{c',r})$.
- **Thresholded sum-to-one**: negative weights are clamped to zero and the remaining positive weights are divided by their sum, producing a sparser normalization with exact zeros.

When enabled, this normalization ensures that the Complexity Layer outputs remain non-negative and sum-bounded, which in turn guarantees that all factors in the tensor product of the Core Mesh (see Section III-D) are strictly positive. This eliminates the vanishing-product problem without requiring a nonlinear activation function in the Complexity Layer, thereby preserving the pure linear HOSVD-based tensor-product structure of the TS fuzzy model. Weight normalization in the Complexity Layer is therefore the recommended approach for ensuring stable gradient flow, as it avoids introducing nonlinearities that would obscure the underlying HOSVD-based tensor-product structure.

It should also be noted that this kind of normalization does not change the structure of the model in a mathematical sense, since the gradients will flow back from the loss function at the output through the normalized weights to the raw weights. At inference time, the normalized weights can substitute for the raw weights without any further need for normalization.

2) Parameterizing the Resolution Layer: The activation functions of the Resolution Layer satisfy $\sum_{r=1}^{R_n} s_{n,r}(x_n, \mathbf{g}_n) = 1$ for all x_n. Only the center positions are learnable; they are parameterized as a base value plus cumulative softplus gaps to guarantee strict ascending order:

$$g_{n,r} = g_{n,0} + \sum_{k=1}^{r-1} \left(\text{softplus}(\tilde{\delta}_k) + \epsilon \right), \tag{23}$$

where $\tilde{\delta}_k$ are unconstrained learnable parameters and $\epsilon > 0$ is a small constant. The left and right distances are derived from neighboring centers: $\delta^L_{n,r} = g_{n,r} - g_{n,r-1}$ and $\delta^R_{n,r} = g_{n,r+1} - g_{n,r}$.

3) Optional Activation Function in the Complexity Layer: Although not a necessary component of the architecture, an optional nonlinear activation function σ and learnable bias $\mathbf{b}_n$ may be introduced in the Complexity Layer. When used, for complexity cell c the output becomes:

$$f^C_{n,c}(x_n) = \sigma \left(\sum_{r=1}^{R_n} u_{n,r,c}\, s_{n,r}(x_n, \mathbf{g}_n) + b^{(n)}_c \right). \tag{24}$$

However, when weight normalization (Section V-D1) is enabled, the activation function in the Complexity Layer can be omitted entirely. Experimental results confirm that weight normalization alone yields equivalent performance to using a softplus activation without normalization. The preferred configuration is therefore to use weight normalization without an activation function in the Complexity Layer, as this preserves the direct correspondence with the classical polytopic formulation widely used in modeling and control.

4) Firing Normalization: When firing normalization is enabled, the raw strengths of the Core Mesh are divided by their sum to form a partition of unity:

$$\hat{r}_{c_1,c_2,\ldots c_N} = \frac{[\mathcal{F}^{CM}(\mathbf{x})]_{c_1,c_2,\ldots c_N}}{\sum_{c'_1,c'_2,\ldots c'_N} [\mathcal{F}^{CM}(\mathbf{x})]_{c'_1,c'_2,\ldots c'_N} + \epsilon}. \tag{25}$$

This mirrors the standard TS fuzzy inference normalization step and keeps output magnitudes bounded regardless of the number of input dimensions.

5) Post-Mesh Smoothing: Independently of the above, an optional smoothing function (e.g., softplus) may be applied after the tensor-product summation in the Output Layer. This does not affect the internal tensor-product structure and can improve performance in classification tasks.

VI. CONCLUSION

In conclusion, the Neural Mesh establishes the foundation of a new class of AI architectures referred to as Logical Synthesis Models (LSMs), designed to complement rather than replace Large Language Models. By combining learnability with structured and interpretable representations, the Neural Mesh provides a framework that naturally supports mathematical analysis, verification, synthesis, and control. Inspired by Tensor Product-based modelling and by the functional role of the cerebellum in biological intelligence, the proposed architecture bridges machine learning with engineering modelling, design and control methodologies. This integration opens new opportunities for developing AI systems that not only learn from data but also support transparent reasoning, provable properties, and the systematic design and control of complex physical and abstract systems.

The proposed Neural Mesh is mathematically equivalent to the TP model, which is widely used in system modelling, control design, and controller synthesis. However, from a structural perspective, the Neural Mesh represents the most influential TP-model components — those manipulated by TP model transformation to achieve desirable modelling and control-theoretic properties — as neural connections and trainable parameters. As a result, these components can be

manipulated through neural-network training procedures, enabling functionalities similar to those achieved by TP model transformation already during the system-identification phase.

This capability of the Neural Mesh opens a new research direction that aims to combine advanced neural-network training methodologies with the TP-model properties traditionally determined through TP model transformation. Such an approach may enable the direct learning of Neural Mesh or TP-model structures and characteristics tailored to specific modelling, optimization, and controller-synthesis objectives, reducing the need for subsequent offline TP model transformation procedures.

REFERENCES

[1] P. Baranyi, Y. Yam, and P. Várlaki, *Tensor Product Model Transformation in Polytopic Model Based Control*, ser. Automation and Control Engineering. CRC Press, Taylor & Frances Group, March 2017, ISBN 9781138077782 - CAT K34341.

[2] P. Baranyi, *TP-Model Transformation Based Control Design Frameworks*, ser. Control Engineering. Springer, July 2016, ISBN 978-3-319-19605-3.

[3] P. Baranyi, *Dual-Control-Design: TP and TS Fuzzy Model Transformation Based Control Optimisation and Design*, ser. Topics in Intelligent Engineering and Informatics (TIEI, volume 17). Springer, December 2023.

[4] P. Baranyi, "TP model transformation as a way to LMI-based controller design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 2, pp. 387–400, 2004. [Online]. Available: DOI: 10.1109/TIE.2003.822037

[5] P. Baranyi, "The generalized TP model transformation for T–S fuzzy model manipulation and generalized stability verification," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 4, pp. 934–948, 2014. [Online]. Available: DOI: 10.1109/TFUZZ.2013.2278982

[6] P. Baranyi, "Extracting LPV and qLPV structures from state-space functions: A TP model transformation based framework," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 3, pp. 499–509, 2020. [Online]. Available: DOI: 10.1109/TFUZZ.2019.2908770

[7] P. Baranyi, "Relaxed TS fuzzy model transformation to improve the approximation accuracy/complexity tradeoff and relax the computation complexity," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 9, pp. 5237–5247, 2024. [Online]. Available: DOI: 10.1109/TFUZZ.2024.3418509

[8] P. Baranyi, "Transition between TS Fuzzy models and the associated convex hulls by TS Fuzzy model transformation," *IEEE Transactions on Fuzzy Systems*, vol. 32, no. 4, pp. 2272–2282, 2024. [Online]. Available: DOI: 10.1109/TFUZZ.2023.3348160

[9] P. Baranyi, "How to vary the input space of a T–S fuzzy model: A TP model transformation-based approach," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 2, pp. 345–356, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2020.3038488

[10] P. Baranyi, "On the crucial role of antecedent-consequent structure selection in LMI-based PDC control of TS fuzzy systems," *IEEE Transactions on Fuzzy Systems*, no. accepted, p. in print, 2026. [Online]. Available: DOI: 10.1109/TFUZZ.2026.3678634

[11] K. Tanaka and H. Wang, *Fuzzy control systems design and analysis: a linear matrix inequality approach*. John Wiley and Sons, Inc., U.S.A., 2001.

[12] C. Scherer and S. Weiland, "Linear matrix inequalities in control," *Lecture Notes, Dutch Institute for Systems and Control*, Delft, The Netherlands, 2000. [Online]. Available: http://www.cs.ele.tue.nl/sweiland/lmi.htm

[13] Y. Qui, J. H. Park, Park, C. Hua, and X. Wang, "Stability analysis of time-varying delay t–s fuzzy systems via quadratic-delay- product method," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 1, pp. 129–137, January 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3141237

[14] A. Ait Ladel, A. Benzaouia, R. Outbib, and M. Ouladsine, "Integrated state/fault estimation and fault-tolerant control design for switched t–s fuzzy systems with sensor and actuator faults," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 8, pp. 3211–3223, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2021.3107751

[15] Z. Sheng, C. Lin, B. Chen, and Q.-G. Wang, "Stability and stabilization of t–s fuzzy time-delay systems under sampled-data control via new asymmetric functional method," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 9, pp. 3197–3209, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2023.3247030

[16] H.-Y. Sun, H.-G. Han, J. Sun, H.-Y. Yang, and J.-F. Qiao, "Security control of sampled-data t–s fuzzy systems subject to cyberattacks and successive packet losses," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 4, pp. 1178–1188, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3197312

[17] Y. Wang, C. Hua, and P. Park, "A generalized reciprocally convex inequality on stability and stabilization for t–s fuzzy systems with time-varying delay," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 722–733, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3187180

[18] Y. Ren, D.-W. Ding, Q. Li, and X. Xie, "Static output feedback control for t–s fuzzy systems via a successive convex optimization algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4298–4309, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3146987

[19] W. Chen, Z. Fei, X. Zhao, and M. V. Basin, "Generic stability criteria for switched nonlinear systems with switching-signal-based lyapunov functions using takagi–sugeno fuzzy model," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4239–4248, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3146975

[20] Y. Kang, L. Yao, and H. Wang, "Fault isolation and fault- tolerant control for takagi–sugeno fuzzy time-varying delay stochastic distribution systems," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 4, pp. 1185–1195, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2021.3053320

[21] X.-P. Xie, Q. Wang, M. Chadli, and K. Shi, "Relaxed observer-based state estimation of discrete-time takagi–sugeno fuzzy systems based on an augmented matrix approach," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 9, pp. 4025–4030, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2021.3129854

[22] X. Xie, C. Wei, Z. Gu, and K. Shi, "Relaxed resilient fuzzy stabilization of discrete-time takagi–sugeno systems via a higher order time-variant balanced matrix method," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 5044–5050, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3145809

[23] X. Xie, F. Yang, L. Wan, J. Xia, and K. Shi, "Enhanced local stabilization of constrained n-ts fuzzy systems with lighter computational burden," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 1064–1070, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3187182

[24] J. Song and X.-H. Chang, "Induced L_∞ quantized sampled-data control for t–s fuzzy system with bandwidth constraint," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 3, pp. 1031–1040, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3193446

[25] Y. Qiu, C. Hua, and Y. Wang, "Nonfragile sampled-data control of t–s fuzzy systems with time delay," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 8, pp. 3202–3210, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2021.3107748

[26] M. S. Aslam, P. Tiwari, H. M. Pandey, and S. S. Band, "Observer-based control for a new stochastic maximum power point tracking for photovoltaic systems with networked control system," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 6, pp. 1870–1884, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3215797

[27] Q. Lv and S. Fang, "A stability analysis method for high-speed magnetically suspended rotors based on tensor product model transformation," *IEEE Transactions on Transportation Electrification*, vol. 11, no. 2, pp. 5201–5210, 2025. [Online]. Available: DOI: 10.1109/TTE.2024.3476674

[28] B.-S. Chen, Y.-Y. Tsai, and M.-Y. Lee, "Robust decentralized formation tracking control for stochastic large-scale biped robot team system under external disturbance and communication requirements," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 2, pp. 654–666, 2021. [Online]. Available: DOI: 10.1109/TCNS.2021.3087621

[29] Y. Wang, S. Xu, and C. K. Ahn, "Finite-time composite antidisturbance control for t–s fuzzy nonhomogeneous markovian jump systems via asynchronous disturbance observer," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 5051–5057, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3160303

[30] S. Tiko and F. Mesquine, "Constrained control for a class of ts fuzzy systems," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 1, pp. 348–353, 2023. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3182775

[31] B. Visakamoorthi, P. Muthukumar, and H. Trinh, "Reachable set estimation for t–s fuzzy markov jump systems with time-varying 8 delays via membership function dependent H_∞ performance," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 4980–4990, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3164799

[32] J. Hu, X. Sun, M. Zhang, and P. Shi, "An offline fuzzy model-predictive control approach using cache," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 10, pp. 4504–4514, 2022. [Online]. Available: DOI: 10.1109/TFUZZ.2022.3154436

[33] P. Várkonyi, D. Tikk, P. Korondi, and P. Baranyi, "A new algorithm for RNO-INO type tensor product model representation," in *Proceedings of the 9th IEEE International Conference on Intelligent Engineering Systems*, 2005, pp. 263–266. [Online]. Available: DOI: 10.1109/INES.2005.1555170

Péter Baranyi received the Ph.D. degree in informatics in 1999 and the Doctor of Science (D.Sc.) degree from the Hungarian Academy of Sciences in 2006.
He is the founder and principal developer of the Tensor Product (TP) model transformation frame-work for control design and optimization. He introduced the framework in 19 papers published in IEEE Transactions journals and three scientific monographs. He is the recipient of several international awards, including the Sigma Xi Investigator Award, the Kimura Award, and the International Dénes Gábor Award. He currently leads the AI NEXT project supported by the Hungarian NKFIH HU-RIZONT program. His current research interests include TP model and convex hull manipulation based control optimization.

Ádám B. Csapó obtained his PhD degree at the Budapest University of Technology and Economics in 2014. Since 2023, he has been teaching and conducting research at Corvinus University of Budapest.
Dr. Csapó's earlier research focused on soft computing tools for developing cognitive infocommunication channels in virtual collaboration environments, with the goal of enabling users to communicate with each other and with their spatial surroundings in novel and effective ways. He has also contributed to the development of a commercial desktop VR platform designed for both educational and industrial applications. More recently, his work has centered on exploratory data analysis and model transformation methods that enhance interpretability and explainability. Dr. Csapó has co-authored more than 100 publications, including two books and over 40 journal articles.

Design and Numerical Analysis of 45-degree Polarization Rotator for Quantum Cryptography

Salwa Marwan Salih, and Shelan Khasro Tawfeeq

***Abstract*—Polarization manipulation in visible wavelengths has an important role in many applications, such as free-space quantum communication and quantum cryptography. In this study, a polarization rotator based on a tilted slot waveguide by 45-degree is proposed and designed. Its principle is to rotate a +45°/-45° input polarized light to TM0/TE0 output polarized light. The simulation results show an extinction ratio of 45.23 (41.17) dB and insertion loss below 0.62 dB at 700 nm wavelength. The bandwidth for an extinction ratio higher than 20 dB is 40 nm. In addition, the device tolerance is analyzed.**

***Index Terms*—Polarization rotator, hybrid modes, visible wavelength, extinction ratio, 45-degree**

I. INTRODUCTION

Integrated photonics holds great promise for the advancement of quantum communication devices in terms of their complex nature, resilience, and scalability [1]. A particular area of focus within this field is silicon photonics, which serves as a prominent platform for quantum photonic technologies [2]. Further benefits it offers miniaturization, low-cost device manufacturing, and compatibility with CMOS microelectronics [3].

Polarization control is essential in numerous applications such as telecommunication, quantum cryptography based on photon polarization encoding, and biosensing [4-7]. Photonic integrated circuits (PICs) have faced significant challenges because of the high confinement and large index contrast that often cause waveguides and devices to be polarization-dependent [8]. The high index contrast, on the other hand, adds significant birefringence to the devices, resulting in polarization-dependent losses or polarization wavelength characteristics [9,10]. As a result, polarization diversity devices such as polarization beam splitters and polarization rotators are required to manipulate the polarization states [11, 12].

The conventional Polarization rotator (PR) waveguides allow the conversion between the orthogonal transverse electric and magnetic modes (TE-TM) [13]. Different structures based on mode evolution, mode hybridization, and surface plasmon effect mechanisms have been reported in [14-16]. Recently, PICs have significant development at a visible wavelength that demands polarization control for chip-scale atomic systems and free-space quantum communication [17]. However, a reduction in wavelength implies a reduction in waveguide size, which requires high manufacturing precision [18]. In quantum communication subsystems, including quantum cryptography repeaters, transmitter and receiver chips, polarization control components are crucial [19, 20]. In addition, shorter wavelength operation would enable polarization rotators more compatible with a variety of single photon emitter sources and several free space transmission systems [21,22]. To the best of our knowledge, few works introduced ±45 polarized light rotators that are very important in QKD protocols. Most of the structures mentioned in the introduction and literature operate at 1550 nm wavelength, while our designed structure operates in the visible region.

The position of the slot has a significant effect on the rotation angle of the polarization rotator. For conventional rotation by 90 degree, the slot position was at 46 nm shifted from the upper corner of Gallium-Phosphide waveguide [12]. While in this structure, the slot is positioned at 19.7 nm.

Salwa Marwan Salih with University of Baghdad, Al-Khwarizmi College of Engineering (e-mail: salwa.m@kecbu.uobaghdad.edu.iq)

Shelan Khasro Tawfeeq with University of Baghdad, Institute of Laser for Postgraduate Studies (e-mail: shelan.khasro@ilps.uobaghdad.edu.iq)

II. DESIGN PRINCIPLE AND NUMERICAL RESULTS

FIGURE 1(A) shows the 3D view of the proposed structure consisting of three parts. The input and output parts are regular Gallium-Phosphide (GaP) strip waveguides with a thickness of 250 nm. The rotation cross-section of the waveguide is shown in Fig. 1(b), which is formed by a horizontal slot of Hydrogen Silsesquioxane (HSQ) tilted by 45º to break the symmetry of the waveguide. To simplify the structure design, the width of the slot (w_{slot}) was fixed to 20 nm and shifted by 19.7 nm from the upper left corner of the rotator waveguide. The width (w) and the height (h) of the GaP waveguide are 197.5 and 200 nm, respectively. The structure is designed by using COMSOL Multiphysics based on a silicon on an insulator platform and covered by air. The refractive indices of GaP, HSQ, and air are 3.2543, 1.41, and 1, respectively, for a wavelength of 700 nm. The 2D simulation framework is truncated by first-order scattering boundary conditions with an extremely fine mesh element size. While in a 3D framework, first-order scattering boundary conditions, in addition to perfect magnetic conductors, are used.

The physical principle of the proposed structure is to rotate an input linear polarized light of +45º/-45º into TM_0 /TE_0 mode. The amplitude of the input (E_{xin}, E_{yin}) and output (E_{xout}, E_{yout}) modes can be expressed by John's matrix as [23],

$$\begin{bmatrix} E_{xout} \\ E_{yout} \end{bmatrix} = J \begin{bmatrix} E_{xin} \\ E_{yin} \end{bmatrix} \quad (1)$$

DOI: 10.36244/ICJ.2026.2.4

Where J is the general John's matrix for a rotator that is [cosθ -sinθ; sinθ cosθ], θ is the rotation angle. For conventional rotators (θ = 90°), i.e., the Johns matrix is defined as [0 -1; 1 0]. This means the fundamental mode TE_0 represented by the vector [1;0] will be completely rotated by 90° into TM_0 mode represented by [0;1]. The same process is applied for rotating the input TM_0 mode to TE_0 mode.

For a special case of an incoming input of 45° linear polarized light represented by 1/√2 [1;1]. To obtain an output of TM_0 mode, the rotation angle should be (θ =45°) with J=1/√2 [1 -1; 1 1]. Accordingly, when a +45° polarized light is launched into the proposed waveguide, two hybrid orthogonal modes are excited with electric field components $E_x < E_y$ for the first mode and $E_x > E_y$ for the second at half of the conversion length (L_c). After complete propagation through the waveguide with an integer number of L_c, it can be completely converted to TM_0 mode. The conversion length (L_c) can be expressed as [24]

$$L_c = \frac{\lambda}{2(n_{eff1} - n_{eff2})} \qquad (2)$$

Where λ is the operating wavelength and n_{eff1} and n_{eff2} are the effective refractive indices of the two hybrid modes.

The rotation angle of the optical axis (φ) caused by modal hybridity is defined as [24]

$$\tan\varphi = \frac{\iint n(x,z)E_x^2(x,z)\,dx\,dz}{\iint n(x,z)E_y^2(x,z)\,dx\,dz} \qquad (3)$$

Where n(x,z) is the refractive index distribution and $E_x(x,z)$ and $E_y(x,z)$ are the horizontal and vertical electrical field components of two orthogonal hybrid modes.

Fig.1(c, d) shows the magnetic field distribution of two hybrid fundamental orthogonal modes at half of L_c. The first mode is with an angle of -67.47° relative to a normal line at the xz-plane, and the second mode is with an angle of 22.49°.

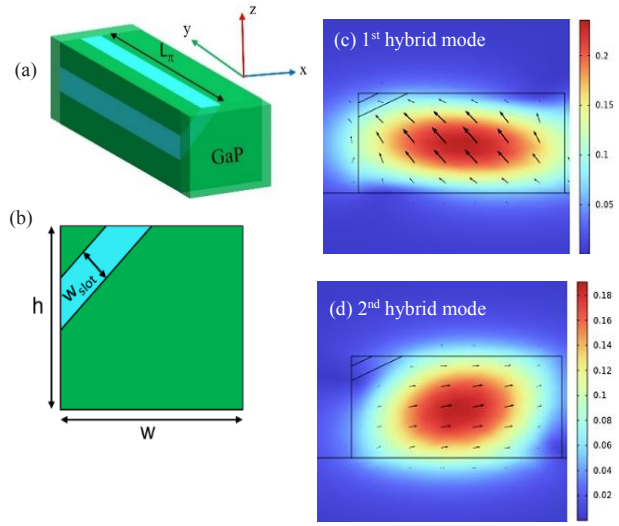


Fig. 1 (a) 3D-view of the polarization rotator (b) cross-section of the rotation region with w and h as the width and the height of the waveguide (c) magnetic field distribution of 1st hybrid mode (d) magnetic field distribution of 2nd hybrid mode. Arrows represent the electric field direction.

The purpose of the structure is to design a polarization rotator that provides a rotation angle of 22.5° at half of L_c so that the input polarized light will be rotated by 45° at L_c. By choosing the appropriate dimensions (w=197.5 nm and h=200 nm), two orthogonal fundamental modes can be fully hybridized as shown in Fig.1 (c and d). The polarization of an incoming +45° polarized light is therefore rotated by 45° into TM_0 mode. In addition, the structure is investigated for rotating -45° input polarized light into TE_0 mode. In Fig.2, the electric field distribution of E_x and E_z components of +45/-45 input polarized light along the rotator waveguide is observed.

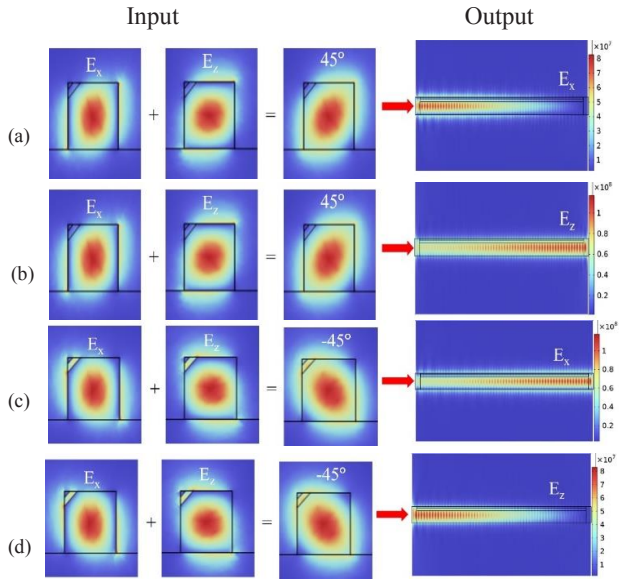


Fig. 2 The electric field distribution of E_X and E_Z components along the rotator waveguide. (a) E_X output when +45° as input (b) E_Z output when +45° as input (c) E_X output when -45° as input (d) E_Z output when -45° as input

The performance of the polarization rotator for input +45°/-45° polarized light is determined by the Polarization Conversion Efficiency (PCE), Extinction Ratio (ER), and Insertion Loss (IL) that can be defined as [25]

$$PCE = \frac{P_{TM-out}}{P_{TE-out} + P_{TM-out}} \qquad (3)$$

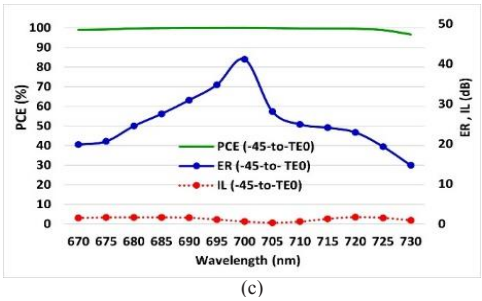
$$ER = 10\log\left(\frac{P_{TM-out}}{P_{TE-out}}\right) \qquad (4)$$

$$IL = -10\log\left(\frac{P_{TM-out}}{P_{in}}\right) \qquad (5)$$

Where $P_{TM\text{-}out}$ and $P_{TE\text{-}out}$ are the output power for TE and TM modes, respectively, and P_{in} is the input power, which is represented by (P_x+P_z).

The wavelength dependence transmission spectra, PCE, ER, and IL of the polarization rotator are illustrated in Fig. 3. The solid lines (blue and red) shown in Fig.3-a represent the transmission for the rotated input light of +45°/-45° polarization to TM_0/ TE_0, respectively. While Fig.3 (b and c) shows the PCE

of 99.99% with a maximum ER 45.23 (41.17) dB for the rotation from +45º/-45º linear polarized light to TM_0/TE_0, respectively. The IL is 0.62 dB at 700 nm wavelength. The bandwidth achieved for ER above 20 dB is 40 nm.
Table (1) shows a comparison between the designed and other structures mentioned in the literature.

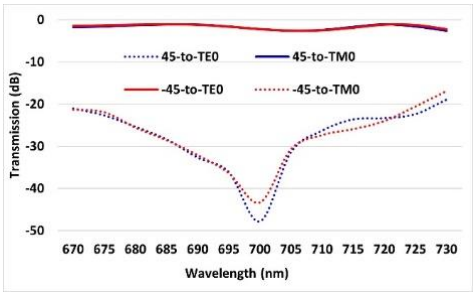


(c)

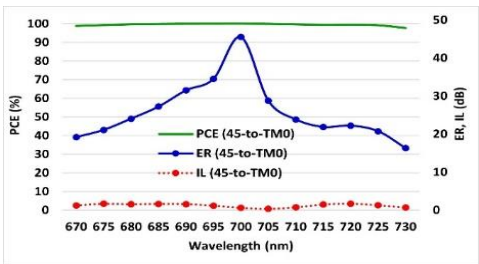
Fig.3 Continued

TABLE I:
A COMPARISON BETWEEN THE DESIGNED STRUCTURE AND VARIOUS PRS

Structure	Input-to-output mode	λ(nm)	PER (dB)	IL(dB)	Lc(μm)	B.W(nm)	Simulation /experimental work
45 deg. W-slot dual-stair waveguide [15]	TE (TM)-to-TM (TE)	1550	>17	<0.4	4.3	300	both
Twisted PR [18]	TE (TM)-to-TM (TE)	1550	---	0.09	150	300	both
Asymmetric slot-waveguide [24]	TE (TM)-to-TM (TE)	1550	25.9	1.19	8	77 TE-to-TM 66 TM-to-TE	simulation
Off-axis PR design [17]	TE (TM)-to-TM (TE)	422	20.7	---	111	---	both
Silicon nitride waveguide PR [19]	TM_0 -to-TE1	780	>20	<1	---	100	both
Our work	±45 to TE_0 (TM_0)	700	> 40	0.62	8.5	40 for PER>20	simulation



(a)



(b)

Fig.3 (a) Transmission spectra for +45º/-45º linear polarized light to TM_0 /TE_0. The PCE, ER, and IL for rotation of (b) 45º linear input polarized light to TM_0 mode (c) - 45º linear input polarized light to TE_0 mode

The performance of the device is affected by the variation in dimensions that occur during the fabrication process. These fabrication errors cause degradation in the device performance, such as PCE and ER. Therefore, it is necessary to study the fabrication tolerance of the rotator waveguide. These fabrication errors have been investigated by varying one parameter, i.e., L_c, and keeping the other parameters fixed at their optimum values. The fabrication tolerance analysis for rotating +45º into TM_0 and -45º TE_0 modes for ER higher than 20 dB is within (ΔL_c = 1μm) shown in Fig.4 and Fig. 5, respectively. The variation in the width of the slot (Δw_{slot}), the width (Δw) and the height (Δh) of the waveguide are listed in Table (2) to obtain ER higher than 15.

TABLE II
VARIATION OF THE DESIGNED STRUCTURE DIMENSIONS

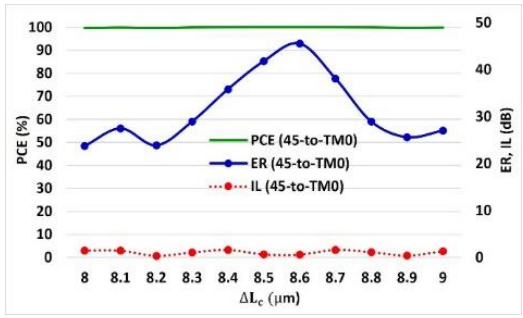
Mode rotation	Variation		
	Δw_{slot}	Δw	Δh
45° to TM_0	±1	±1	±1
-45° to TE_0	-1 to +2	±1	±1

According to Table (2), and Fig. 4 and Fig.5, the results show that the tolerance of Δw_{slot}, Δw and Δh is very tight compared to ΔL_c.

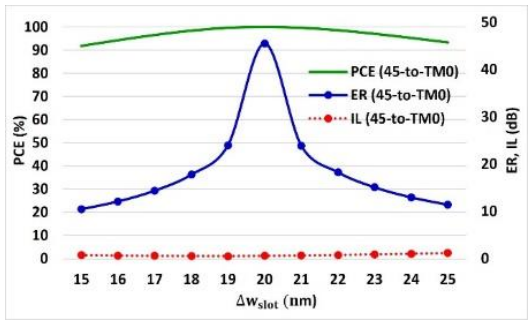
III. Conclusion

A 45-degree polarization rotator based on a tilted slot waveguide is designed. The device aimed to rotate a +45º/-45º

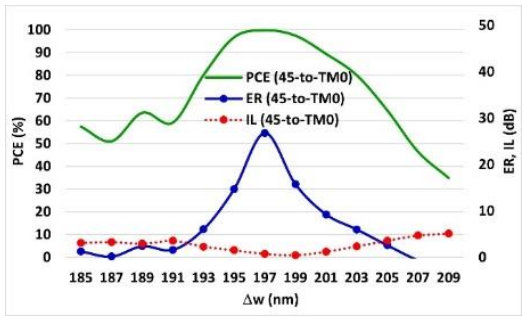
input polarized light by a rotation angle of 45º; to obtain a TM_0/TE_0 polarized output light. The device has ER for the rotation from +45º/-45º linear polarized light to TM_0/TE_0 of 45.23 (41.17) dB, respectively, with IL of 0.62 dB at 700 nm wavelength. Also, the wavelength dependence and fabrication tolerance of the device are analyzed.



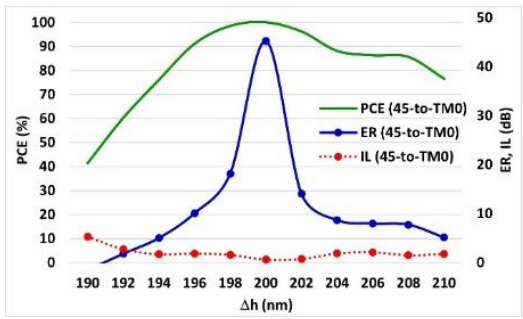
(a)



(b)

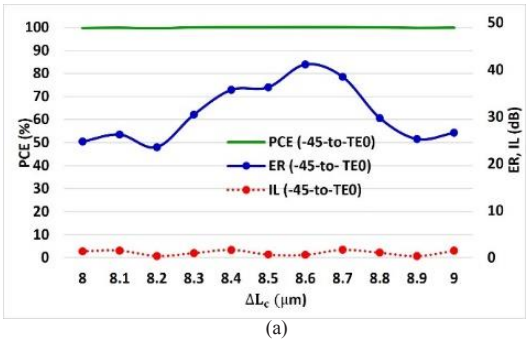


(c)

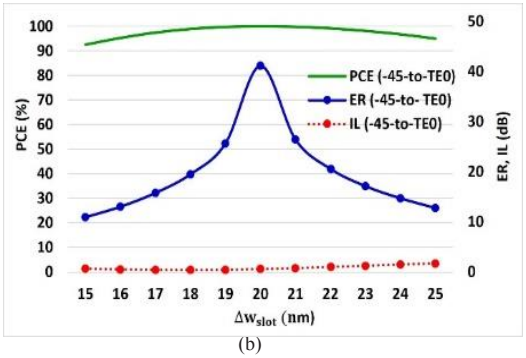


(d)

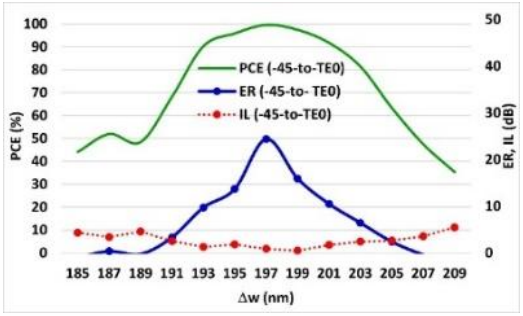
Fig.4 The fabrication error as a function of PCE, ER, and IL for rotating 45º linear polarized light to TM_0 mode (a) ΔL_c (b) Δwslot (c) Δw (d) Δh



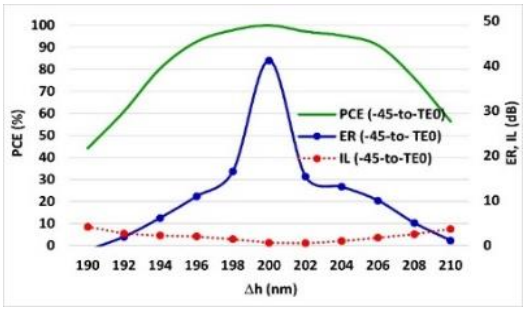
(a)



(b)



(c)



(d)

Fig.5 The fabrication error as a function of PCE , ER and IL for rotating -45º linear polarized light to TE0 mode (a) ΔL_c (b) Δwslot (c) Δw (d) Δh

REFERENCES

[1] Sibson, P., Kennard, J. E., Stanisic, Erven, S., C., O'Brien, J. L., Thompson, M. G.," Integrated silicon photonics for high-speed quantum key distribution," *Optica*, vol. 4, No. 2, pp. 172–177, 2017. **DOI**: 10.1364/OPTICA.4.000172

[2] Al-mfrji, A. A., Tawfeeq S. K., Fyath, R.S.," Modeling and Analysis of a Miniaturized Ring Modulator Using Silicon-Polymer-Metal Hybrid Plasmonic Phase Shifter. Part I: Theoretical Framework,". I*nternational Journal of Optics and Applications*, vol. 5, no. 4, pp. 121–132, 2015. **DOI**: 10.5923/j.optics.20150504.03

[3] Du, Y., Zhu, X., Hua, X., Zhao, Z., Hu, X., Qian, Y., Xiao, X., Wei, K.," Silicon-based decoder for polarization-encoding quantum key distribution," *Chip*, vol. 2, no.1, p. 100 039, 2023. **DOI**: 10.1016/j.chip.2023.100039

[4] Kumar, J., Saxena, V." Cloud Data Security through BB84 Protocol and Genetic Algorithm," *Baghdad Sci. J.*, vol. 19, issue 6, pp. 1445–1453 2022. **DOI**: 10.21123/bsj.2022.7281

[5] Al Hasani, M. H., Al Naimee, K. A. M.," Quantum Key Distribution and Chaos Bandwidth Effects on Impact Security of Quantum Communications," *Iraqi Journal of Science*, vol. 60 no. 6, pp. 1266–1273, 2019. **DOI**: 10.24996/ijs.2019.60.6.10

[6] Patton, R. J., Reano, R. M.," Framework for tunable polarization state generation using Berry's phase in silicon waveguides," *Optics Express*, vol. 28 no. 14, p. 20 845, 2020. **DOI**: 10.1364/OE.384543

[7] Chandra, D., Botsinis,P., Alanis, D., Zunaira Babar, Z., Ng, X.-Z., Hanzo, L., " On the Road to Quantum Communications,", *Infocommunications Journal*, vol. VIX, no. 2, 2022. **DOI**: 10.36244/ICJ.2022.3.1

[8] Gao, L., Huo, Y., Zang, K., Paik, S., Chen, Y., Harris, J. S., Zhou, Z.," On-chip plasmonic waveguide optical waveplate," *Scientific Reports*, vol. 5, p. 15 794, 2015. **DOI**: 10.1038/srep15794

[9] Wang, J., Xiao, J., Sun, X.," Design of a broadband polarization rotator for silicon-based cross-slot waveguides," *Applied Optics*, vol. 54 no. 12, pp. 3805–3810, 2015. **DOI**: 10.1364/AO.54.003805

[10] Xiao, J., Xu, Y., Wang, J., Sun, X., "Compact polarization rotator for silicon-based slot waveguide structures," *Applied Optics*, vol. 53, no. 11, pp. 2390–2397, 2014. **DOI**: 10.1364/AO.53.002390

[11] Wang, J." Design of a compact polarization beam splitter for silicon-based cross-slot waveguides," *Journal of Modern Optics* vol. 67, no. 15, pp. 1277–1284, 2020. **DOI**: 10.1080/09500340.2020.1842928

[12] Salih, S. M., Tawfeeq, S. K." Integrated gallium phosphide-waveguide polarization rotator based on rotating a horizontal slot by 45 degree for 700 nm wavelength," *Journal of Optics*, vol. 53, pp. 2168–2173, 2024. **DOI**: 10.1007/s12596-023-01420-6

[13] Ding, Y., Bacco, D., Llewellyn, D., Faruque, I., Paesani, S., Galili, M., Laing, A., Rottwitt, K., Thompson, M., Wang, J., and Oxenløwe, L. K.," Silicon Photonics for Quantum Communication," in *Proceeding of 21st International Conference on Transparent Optical Networks (ICTON)*, 18994402 2019.

[14] Giordani, T., Hoch, F., Carvacho, G., Spagnolo, N., Sciarrino, F., " Integrated photonics in quantum technologies," *La Rivista del Nuovo Cimento*, vol. 46, pp. 71–103, 2023. **DOI**: 10.1007/s40766-023-00040-x

[15] Hui, Z., Wen, X., Pan, D., Han, D., Li, T., Gong, J.," Ultrabroadband polarization rotator based on 45 deg W-slot dual-stair waveguide covering S + C + L + U bands," *Optical. Engineering*, vol. 60, no. 12, p. 127 103, 2021. **DOI**: 10.1117/1.OE.60.12.127103

[16] Khaleel, F. A., Tawfeeq, S. K." On the use of Aluminium as a plasmonic material in polarization rotators based on a hybrid plasmonic waveguide,"*Iraqi Journal of Laser*, vol. 20 issue, 2, pp. 23–41, 2021.

[17] Sneh, T., Hattori, A., Notaros, M., Corsetti, S., Notaros, J." Design of Integrated Visible-Light Polarization Rotators and Splitters," *Proceeding in Frontiers in Optics + Laser Science*, JTu5A.48 2022.

[18] Hou, Z.-S, Hou, Z.-S., Xiong, X., Cao, J.-J., Chen, Q.-D., Tian, Z.-N., Ren, X.-F., Sun, H.-B.," On-Chip Polarization Rotators," *Advanced Optical Materials*, vol. 7 no. 10, p. 1 900 129, 2019. **DOI**: 10.1002/adom.201900129

[19] Gallacher, K., Griffin, P.F., Riis, E., Sorel, M., Paul, D. J.," Silicon nitride waveguide polarization rotator and polarization beam splitter for chip-scale atomic systems," A*PL Photon*, vol. 7, p. 046 101, 2022. **DOI**: 10.1063/5.0077738

[20] Jaber, M. S., Tawfeeq, S. K., Fyath, R. S." Design Investigation of 4 x 4 Nonblocking Hybrid Plasmonic Electrooptic Switch," *Photonics*, vol. 6 no. 47, 2019. **DOI**: 10.3390/photonics6020047

[21] Khaleel, F. A., Tawfeeq, S.K.," The spontaneous emission performance of a quantum emitter coupled to a hybrid plasmonic waveguide with specified output polarization for on- chip plasmonic single-photon source,". *Photonics and Nanostructures – Fundamentals and Applications*, vol. 45, p. 100 925, 2021.
https://doi.org/10.1016/j.photonics.2021.100925

[22] Khaleel, A. I., Tawfeeq, S.K.," Key rate estimation of measurement-device-independent quantum key distribution protocol in satellite-earth and intersatellite links," *International Journal of Quantum Information*, Vol. 16, no. 3, p. 1 850 027, 2018. **DOI**: 10.1142/S0219749918500272

[23] Pedrotti, F. L., Pedrotti, L. M., Pedrotti, L. S.: "Introduction to optics," (Addison-Wesley, 2006).

[24] Zhang, Z., Tsuji, Y., Masashi, E., . Chen, C.-P, Anada, T., " Design of Polarization Rotator Based on Asymmetric Slot-Waveguide," in *Proceeding Asia Communications and Photonics Conference*. M4A.111, 2019.

[25] Xie, A., Zhou, L., Chen, J., Li, X.: Efficient silicon polarization rotator based on mode-hybridization in a double-stair waveguide. *Opticss Express*, vol. 23, no.4, pp. 3960–3970, 2015. **DOI**: 10.1364/OE.23.003960

Salwa Marwan Salih earned her B.Sc. degree from the College of Engineering, University of Baghdad, Baghdad, Iraq, in 2013 and the M.Sc. and Ph.D. degrees from the Institute of Laser for Postgraduate Studies, University of Baghdad, in 2019 and 2024, respectively. At present, she is a lecturer at Al-Khwarizmi College of Engineering, University of Baghdad. Her research interest is focused on integrated photonics and optical communication systems.

Shelan Khasro Tawfeeq received a B.Sc. degree from the College of Engineering, University of Baghdad, Baghdad, Iraq, 1990, and the M.Sc. and Ph.D. degrees from the Institute of Laser for Postgraduate Studies, University of Baghdad, in 2001 and 2006, respectively. She is currently an Assistant Professor with the Institute of Laser for Postgraduate Studies, University of Baghdad, and head of Quantum Optics and Electronics Group where she is involved in research in quantum information science and optical communications systems.

Supply Chain Logistics Demand Modeling and Abnormal Warning Based on Autoformer

Chuanchuan Teng, and Xizhuo Han*

Abstract—**Current methods have limited capabilities in accurately modeling and dynamically warning of anomalies in port supply chain logistics (SCLs) scenarios where long-term trends in logistics demand (LD) coexist with short-term disturbances. To improve the accuracy of LD forecasting and anomaly warning (AW), and enhance the stability and responsiveness of the supply chain (SC), this paper constructed a multi-scale demand forecasting and AW model based on Autoformer. First, multi-source logistics data was integrated and combined with the Autoformer trend-disturbance decomposition mechanism to achieve structured modeling of non-stationary demand sequences. Then, a multi-scale attention path was designed to extract periodic information at the daily, monthly, and quarterly levels, respectively, to enhance the model's perception of multi-frequency features. Finally, a residual-driven anomaly scoring function and a dynamic threshold strategy were combined to construct a multi-level AW mechanism to achieve accurate identification of logistics risk events of different intensities. Experimental results show that in LD modeling, the Autoformer achieves a final weighted average percentage wrror (WAPE) and root mean square error (RMSE) of 0.105, and a matching degree of over 0.900 for the changing trend of supply chain logistics demand (SCLD) for a 200 twenty-feet equivalent unit (TEU) SC. In terms of AW, the Autoformer achieves average false negative rate (FNR) and false positive rate (FPR) of 12.0% and 7.3%, respectively, with an average overall delay of 1.3 hours. The weighted Kappa coefficient for consistency in AW levels is 0.844. The results show that the proposed Autoformer model, on the port logistics dataset used, demonstrates certain improvements in demand forecasting and anomaly detection compared to the selected baseline model, providing a feasible solution reference for forecasting and early warning tasks in intelligent supply chain management.**

Index Terms—**Supply Chain Logistics, Abnormal Warning, Autoformer Model, Multi-Scale Attention Mechanism, Trend Perturbation Decomposition**

I. INTRODUCTION

WITH the deep penetration of information technologies such as the Internet of Things, 5G communication and cloud computing into the logistics field, the modern supply chain has evolved into a highly information-based and interconnected complex system. Real-time information interaction between logistics nodes and the fusion processing of multi-source heterogeneous data have become the key foundation for supporting accurate decision-making and agile response in the supply chain. Against this background, port logistics demand (PLD) exhibits typical characteristics of multivariability, high dynamism and multi-scale, with complex data dimensions and intertwined influencing factors [1], [2]. Introducing cutting-edge deep learning methods suitable for high-dimensional time series data processing into supply chain logistics modeling is not only a natural extension of the method, but also an inherent requirement for the digital transformation and upgrading of the industry. Existing risk warning models struggle to accurately capture the correlations between key indicators such as cargo throughput, ship arrival frequency, and warehouse turnover [3], [4], [5]. Faced with both long-term trend fluctuations and short-term sudden fluctuations in PLD, these models struggle to address their non-stationarity, the coexistence of long-term and short-term trends, and sudden disruptions. This can lead to inaccurate forecasts and delayed AWs, severely restricting the rational allocation of port logistics resources and the timeliness of emergency responses.

This paper combines the Autoformer model [6]with the structural characteristics of SC port logistics scenarios to construct an enhanced prediction and anomaly detection (AD) framework. By constructing an adaptive trend decomposition module and a multi-scale attention mechanism (MSAM), this paper achieves a high-precision forecasting and dynamic AW of PLD, which has important theoretical value and practical significance for improving the logistics response efficiency and risk prevention and control level of ports.

The main contributions of this paper are reflected in the integration and scenario-specific improvements of existing time-series modeling components tailored to the characteristics of SCLs scenarios:

1) A "trend-disturbance dual-path adaptive decomposition architecture" for the non-stationary demands of port logistics is proposed. This architecture improves the residual decomposition problem of the original Autoformer under strong sudden disturbances by using a learnable sliding average window and residual paths.

2) A nested multi-scale attention fusion mechanism is designed. Through learnable scale weights, it achieves adaptive extraction and fusion of multi-level periodic features at the daily, monthly, and quarterly levels, overcoming the limitations of fixed scale division.

School of Economics and Management, Cangzhou College of Transportation, Cangzhou City, Hebei Province, China
* Corresponding Author, e-mail: hanxizhuo0313@163.com

DOI: 10.36244/ICJ.2026.2.5

3) A dynamic residual scoring and multi-level threshold early warning system is constructed. Combining local statistics of the sliding window and duration judgment, it achieves graded and sensitive responses to anomalies of different degrees in logistics scenarios.

II. Related Work

In recent years, the informatization and intelligent transformation of SC logistics systems has become a consensus in academia and industry. Real-time monitoring of logistics processes, multi-source data fusion and intelligent decision support are all inseparable from the support of information and communication technology (ICT) and artificial intelligence methods [7]. In particular, with the implementation of concepts such as industrial internet and digital twins, logistics systems have gradually built a two-way mapping of "physical process-information space", which provides rich application scenarios and data foundation for data-driven methods such as deep learning [8], [9]. Some scholars used statistical and neural network models to model SCLs demand, namely, the combination of long short-term memory (LSTM) and auto-regressive integrated moving average (ARIMA) models, as well as neural network and grey prediction models, to improve forecasting accuracy [10]. Zhu Junlin built a data-driven prediction framework by enhancing the LSTM structure, which not only significantly reduced the RMSE but also achieved a 95% true positive rate in AD, demonstrating strong responsiveness [11]. In addition, unsupervised methods such as isolation forests have also been used for AD, especially in practical scenarios where labels are scarce [12]. Yipeng Chen focused on the changes in prediction accuracy at different time granularities and proposed a multi-granularity joint optimization framework to improve resource coordination issues at all levels of the SC [13]. Xie Luqiang proposed a warning model that combines rough sets with radial basis function neural networks to address the hierarchical complexity of cruise ship construction logistics needs, improving the prediction efficiency and emergency response capabilities in specific scenarios [14]. These studies have provided a wealth of methodological support for SCLs' demand forecasting and anomaly identification, but most methods have the limitation of insufficient ability to model long-term dependent features.

To further improve the ability to model long-term dependent features, some researchers have begun exploring modeling methods based on Autoformers, leveraging their built-in autocorrelation mechanisms and decomposition structures to enhance their ability to capture periodicity and trends. Ma Xiangkai, addressing the limitations of traditional Autoformers in modeling multi-scale features, proposed applying multi-scale convolution operations and a temporal encoding mechanism, and constructed an Autoformer model that significantly improved RMSE performance in multi-domain prediction tasks [15]. Cao Danyang designed a more efficient Autoformer structure and added a position embedding mechanism to improve the model training speed and generalization ability, achieving improvements in multiple evaluation indicators [16]. Based on Autoformer, these studies have preliminarily verified its potential in processing complex SC time series data. However, existing multi-scale Transformer methods mostly employ fixed or predefined scale divisions, making it difficult to adaptively capture the complex patterns of interwoven daily, monthly, and quarterly cycles in port logistics. There is also limited research on decoupling the differentiated impacts of trends and disturbances at different scales. While current decomposition-based Transformer models can separate trend and seasonal terms, in port logistics scenarios, short-term sudden disturbances are easily smoothed out or remain in the trend term, leading to decreased anomaly detection sensitivity. Furthermore, most residual-driven anomaly detection methods rely on global or static thresholds, making it difficult to adapt to changes in local statistical characteristics under dynamic fluctuations in logistics demand, and lacking the ability to provide tiered early warnings for anomalies of varying severity.

III. Supply Chain Logistics Demand Modeling and Abnormal Warning Based on Autoformer

A. Supply Chain Multi-Source Heterogeneous Data Fusion

To address the combined impact of historical order volume, inventory turnover, shipping time, and heterogeneous market environments on changing demand in SCLs scenarios, this paper constructs a unified input feature system. First, a multi-source, heterogeneous time series data fusion framework for port logistics scenarios is designed. Through data fusion and standardization, an input dataset with a unified time series structure is constructed.

The multi-source data used in this paper primarily encompasses four dimensions: historical order data, inventory information, shipping time, and market environment indicators. Table I illustrates the original data format.

Table I
Data characteristics of different dimensions

Data type	Feature Name	Time granularity	Unit
Historical order data	Daily order quantity	day	TEU
Inventory data	Current inventory level	day	ton
Shipping time	Average shipping time	hour	hour
Market environment	Consumer Price Index (CPI)	month/day	Index/Boolean
	Purchasing Managers´Index (PMI)		
	Holiday signs		

To achieve a unified input format, this paper resamples all data at a daily granularity, using nearest neighbor filling for monthly data, daily average aggregation for hourly data, and Boolean flags for holidays (0 for non-holidays, 1 for holidays). To address irregular data sampling, transmission delays, and missing values, this paper uses time index alignment and linear

interpolation to align and fill missing values. Using the order data timeline as the primary index, outer joins are used for the remaining dimensions to generate a unified date-aligned data table. Linear interpolation is used for continuous variables:

$$\hat{x}_t = x_t + \frac{(x_{t_2} - x_{t_1})(t - t_1)}{t_2 - t_1},\ t \in (t_1, t_2) \quad (1)$$

x_{t_1} and x_{t_2} are the nearest non-missing values before and after the missing point. t_1 and t_2 are the nearest valid time points before and after the missing value, respectively. Forward filling is used for discrete or Boolean variables. Due to the significant differences in the dimensions and numerical distribution of each variable, directly inputting the model leads to problems such as uneven feature response and unstable gradients. This paper uses Z-score normalization for all continuous features:

$$z_{i,t} = \frac{x_{i,t} - \mu_i}{\sigma_i} \quad (2)$$

Here, μ_i and σ_i are the mean and standard deviation of the feature in the training set, and $x_{i,t}$ signifies the original value of the i-th feature at time t. After normalization, construct the sliding window input sequence $X_t \in \mathrm{R}^{\omega \times d}$ [17], [18]:

$$X_t = [x_{t-\omega+1}, x_{t-\omega+2}, \cdots, x_t]^{\mathrm{T}} \quad (3)$$

The final result is a fixed-length sliding sequence input set $\{X_t, y_t\}$, where y_t is the corresponding target prediction value. To enhance the model's ability to perceive temporal position information and periodic fluctuations, timestamp features are added to the final input, including the week number $Day\ of\ week_t \in \{1,2,3,4,5,6,7\}$, $Month_t \in \{1,2,\cdots,12\}$, and holiday indicator variable $Holiday_t \in \{0,1\}$ at the current time step t. All processed timestamp variables are concatenated column by column to obtain the structured time feature matrix $T_t \in R^m$ corresponding to each time step, where m is the total dimension after encoding. To improve the model's ability to perceive temporal position information, a continuous position encoding based on sine and cosine functions is applied [19], [20]:

$$PE_{(t,2i)} = sin\left(\frac{t}{10000^{2i/d}}\right) \quad (4)$$

$$PE_{(t,2i+1)} = cos\left(\frac{t}{10000^{2i/d}}\right) \quad (5)$$

Finally, this paper concatenates the encoded structured timestamp feature T_t and the continuous position embedding PE_t in the feature dimension to form an enhanced temporal structure feature vector $\tilde{T}_t \in \mathrm{R}^{m+d}$. This vector is input into the model together with the standardized main sequence data.

B. Supply Chain Trend-Disturbance Adaptive Decomposition

SCLs demand that time series contain long-term trends, cyclical fluctuations, and sudden disturbances [21], [22]. Unified encoding methods are prone to reducing the model's time series prediction accuracy and anomaly identification capabilities due to dynamic feature aliasing [23], [24]. In order to achieve effective analysis and prediction of non-stationary supply chain circulation demand, this paper uses the Autoformer model for adaptive trend decomposition. The input multivariate time series is structurally decomposed to construct trend path and interference path structures to achieve feature transfer processing, as shown in Figure 1.

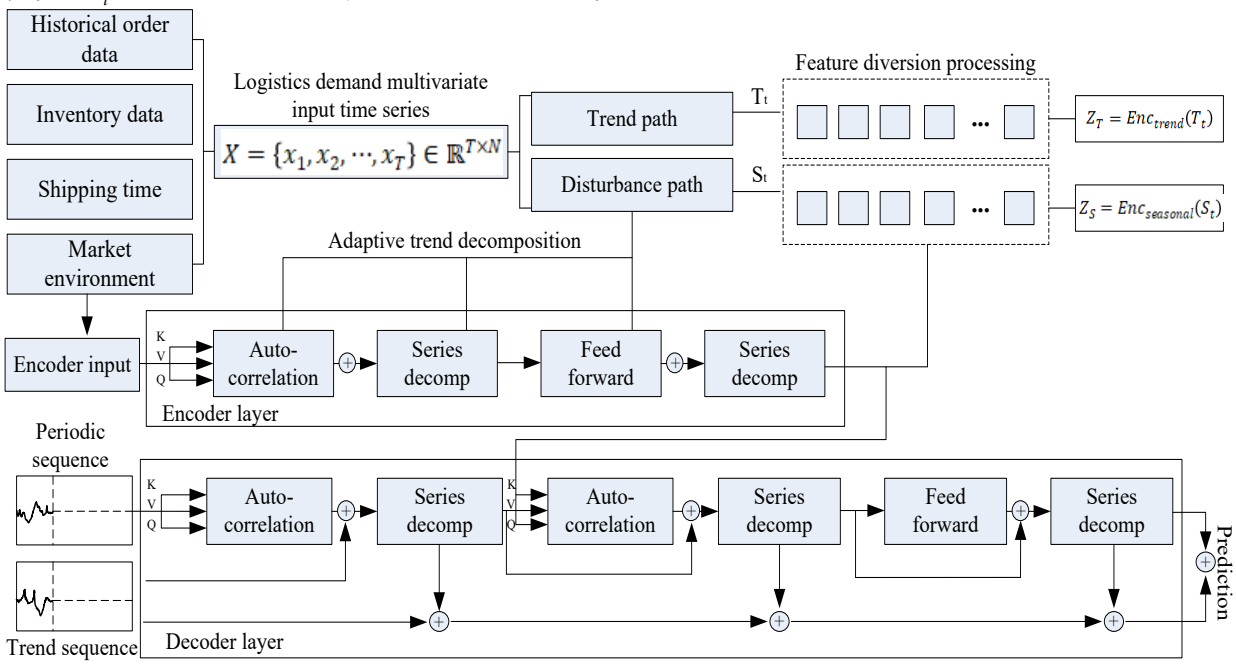


Fig. 1. Adaptive Trend Decomposition and Feature Diversion Modeling

Considering a multivariate input time series $X = \{x_1, x_2, \cdots, x_T\} \in R^{T \times N}$, and the decomposition goal is:

$$X_t = T_t + S_t \quad (6)$$

T_t represents the trend term, which represents the low-frequency trend structure that changes over time; S_t represents the residual term, which contains local perturbations and high-frequency variation components. In order to achieve a learnable decomposition operation, this paper uses a sliding average filter

as an approximate low-pass filter. Given a multivariate time series $X=\{x_1, x_2, \cdots, x_T\} \in R^{T\times N}$, and assuming the half-length of the sliding window is l (l is a positive integer), the trend term T_t at time point t is calculated as follows:

$$T_t=\frac{1}{2l+1}\sum_{i=-l}^{l} x_{t+i},\ t\in[l+1, T-l] \tag{7}$$

In formula, x_{t+i} represents the observation vector at time point $t+i$. The remaining part constitutes the residual term S_t:

$$S_t=X_t-T_t \tag{8}$$

The sliding average window is constructed as a learnable parameter, and the window length and weight are automatically optimized through the training process to achieve dynamic adaptability to different data sources. In the encoder part, two path diversion attention modules are designed, namely the trend path and the perturbation path, to perform structural modeling on T_t and S_t, respectively, to implement a dynamic feature diversion strategy. The trend term has strong long-term dependence and slowly changing characteristics, and it is suitable to use an attention mechanism with a larger receptive field to extract global context information. An autocorrelation-based structural attention extractor is used to replace the standard dot product attention calculation with time-delayed correlation estimation to improve efficiency and long-term modeling capabilities:

$$AutoCoor(Q,K)=\frac{1}{T}\sum_{\tau=0}^{T-1} Q_t \cdot K_{t-\tau} \tag{9}$$

Q and K are the query and key vectors, and the delay τ represents the matching strength at different time offsets. This mechanism enhances the time series's ability to identify periodic and delayed relationships. The decomposed sum is simultaneously fed into the attention sub-network through its respective encoders. At the end of the encoder, this paper applies a fusion module to integrate the modeling results of the two sub-paths. Considering the trend encoding output as Z_T and the perturbation encoding output as Z_S:

$$Z_T=Enc_{trend}(T_t) \tag{10}$$

$$Z_S=Enc_{seasonal}(S_t) \tag{11}$$

Then, performing dynamic weight fusion:

$$Z=\alpha \cdot Z_T+(1-\alpha)\cdot Z_S \tag{12}$$

α is a learnable parameter that adaptively allocates trends based on the time series variance. The final structure is formed and input into the decoder, which outputs the final predicted demand value through inverse reconstruction and dynamic frequency modulation.

C. Multi-scale Attention Mechanism for Supply Chain Management

To capture multi-level cyclical patterns in logistics demand, such as daily, monthly, and quarterly patterns, this paper adds a nested multi-scale attention module after the Autoformer encoder, as shown in Fig. 2.

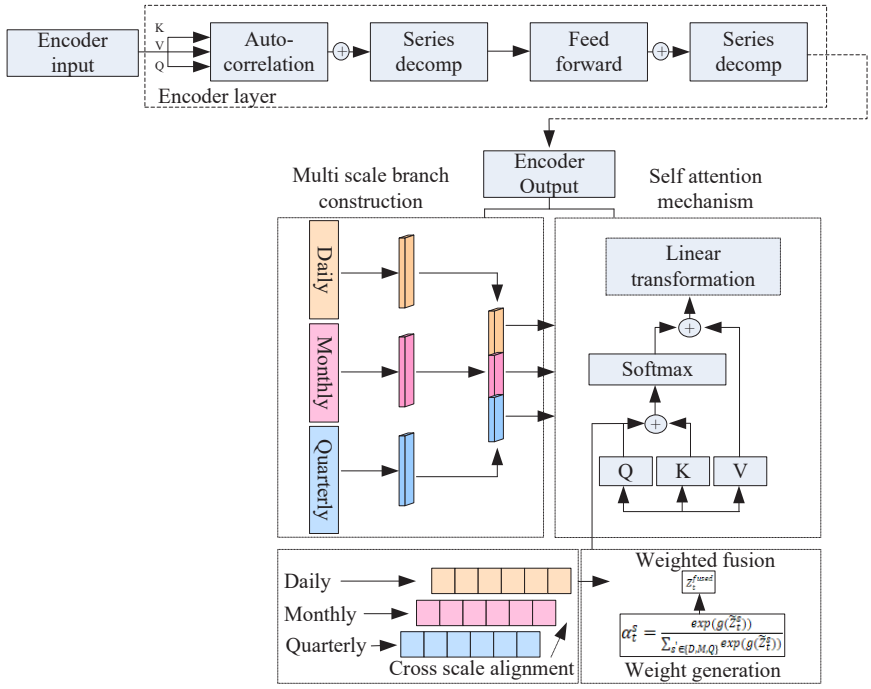


Fig. 2. Multi-scale Attention Module

This paper first reconstructs the long-term trend term T_t and high-frequency residual term S_t output by the decomposition module through multi-scale windows to construct input blocks of varying granularity. Assuming the original sequence length is L, three scale windows are defined:

1) Quarterly scale s_Q, window length l_Q;
2) Monthly scale s_M, window length l_N;
3) Daily scale s_D, window length l_D.

Let the intermediate representation after the trend decomposition module and encoder processing be $H\in R^{T\times N}$.

Based on the sliding window mechanism, the attention input sequences of three sub-paths are constructed respectively at the day level (D), month level (M), and quarter level (Q):

$$H^D=Segment(H,l_D)=\{H_1^D,H_2^D,H_3^D\} \quad (13)$$

$$H^M=Segment(H,l_M)=\{H_1^M,H_2^M,H_3^M\} \quad (14)$$

$$H^Q=Segment(H,l_Q)=\{H_1^Q,H_2^Q,H_3^Q\} \quad (15)$$

Then, corresponding attention modules are constructed for the inputs at the three scales. Each module adopts an improved local-global nested attention structure to capture both local perturbations and long-term dependencies. On each subpath, an independent multi-head self-attention module is used to extract local periodic dependency features:

$$Z^D=MSA_D(H^D) \quad (16)$$

$$Z^M=MSA_M(H^M) \quad (17)$$

$$Z^Q=MSA_Q(H^Q) \quad (18)$$

MSA represents a multi-head attention mechanism based on temporal position embedding to capture local dynamic patterns. Based on the periodic pattern extraction strategy at each scale s_k, a global representation across periods is constructed:

$$P_{s_k}=\sum_{j=0}^{\lfloor L/l_k\rfloor-1} X_{s_k}^{(jl_k)}\in \mathbb{R}^{\lfloor L/l_k\rfloor\times l_k\times N} \quad (19)$$

Then, building a cycle attention mechanism on the cycle dimension:

$$Attention_{global}^{(k)}=Softmax\left(\frac{Q_p K_p^{\mathrm{T}}}{\sqrt{d_k}}\right)V_p \quad (20)$$

In Formula (20), Q_p, K_p, and V_p are the query, key, and value expressions in the periodic block. Due to the different sliding window lengths and step sizes of different scale paths, the lengths of the output sequences are different [25], [26]. To achieve subsequent fusion, the output features of different scales are length-aligned and reconstructed. Assuming the final target sequence length is T', linear interpolation is used to map Z^D, Z^M, and Z^Q back to the unified time axis:

$$\tilde{Z}^s=Align(Z^s,T'),\ s\in\{D,M,Q\} \quad (21)$$

Align represents the scale alignment operation, and outputs the aligned multi-scale feature tensor $\tilde{Z}^s\in\mathbb{R}^{T'\times N}$. A learnable scale fusion attention weight mechanism is employed to model the significance of features at different time scales. For each moment $t\in[1,T']$, three scale fusion weights are generated through the gating network $g(\cdot)$ [27], [28], [29]:

$$\alpha_t^s=\frac{exp(g(\tilde{Z}_t^s))}{\sum_{s'\in\{D,M,Q\}} exp(g(\tilde{Z}_t^{s'}))},\ s\in\{D,M,Q\} \quad (22)$$

The aligned multi-scale feature representations are weighted and fused to generate the final integrated sequence output $Z^{fused}\in\mathbb{R}^{T'\times N}$[30], [31]:

$$Z_t^{fused}=\alpha_t^D\cdot\tilde{Z}_t^D+\alpha_t^M\cdot\tilde{Z}_t^M+\alpha_t^Q\cdot\tilde{Z}_t^Q,\ t\in[1,T'] \quad (23)$$

The output is fed into the decoder part of the Autoformer for final prediction reconstruction and abnormal residual scoring.

D. Dynamic Early Warning Driven by Supply Chain Risks

Based on model prediction errors, this method utilizes dynamic residual distribution modeling and multi-dimensional threshold adjustment strategies to score, identify, and trigger warning signals for abnormal conditions in system operation, providing data-driven risk prevention capabilities for logistics management. After the Autoformer model completes the modeling of trend and disturbance terms, these two types of features are collaboratively restored in the decoding phase to obtain the final prediction value sequence. Assuming that the trend prediction term output by the model is $\hat{T}_t$ and the disturbance prediction term is $\hat{S}_t$, the final demand forecast sequence can be written as:

$$\hat{y}_t=\hat{T}_t+\hat{S}_t,\ t=1,2,\cdots,\hat{H} \quad (24)$$

By retaining the independence of the characteristics of different time dynamic components and aggregating long-term trends and short-term disturbances, the demand forecast expression is achieved. Based on the historical true value y_t and the predicted value $\hat{y}_t$, the time series residual vector e_t is calculated:

$$e_t=y_t-\hat{y}_t \quad (25)$$

This residual sequence is directly used in the subsequent AD scoring mechanism, and its statistical characteristics are used to characterize the degree of deviation between normal and abnormal states. In response to the non-stationary and dynamic structural changes of SC data, this paper adopts a residual-driven dynamic anomaly scoring method. Let e_t be the prediction residual at time point t, and W be the size of the sliding window. The scoring function is defined based on the absolute deviation of the standardized residual:

$$z_t=\frac{e_t-\mu_e}{\sigma_e} \quad (26)$$

μ_e and σ_e are the mean and standard deviation of the residual sequence $\{e_{t-W+1},\cdots,e_t\}$ in the sliding window, respectively:

$$\mu_e=\frac{1}{W}\sum_{i=t-W+1}^{t} e_i \quad (27)$$

$$\sigma_e=\sqrt{\frac{1}{W}\sum_{i=t-W+1}^{t}\left(e_t-\mu_e\right)^2} \quad (28)$$

The residual score function is defined as:

$$S_t=|z_t| \quad (29)$$

Setting the signal trigger threshold θ_t to:

$$\theta_t=\mu_{S_t}+K\cdot\sigma_{S_t} \quad (30)$$

If the current score meets the following conditions, it is considered abnormal:

$$S_t>\theta_t\Rightarrow Anomaly\ at\ t \quad (31)$$

To improve the ability to identify multiple types of anomalies, this paper further designs multi-level judgment rules, combined time continuity and anomaly level distribution, and constructs an AW system based on a scoring function and duration, as shown in Table II:

TABLE II
ABNORMAL WARNING SYSTEM

Abnormal level	Residual rating threshold	Duration	Response Level
Normal	$S_t \leq \theta_t$	No requirements	No alarm
1-Mild abnormality	$\theta_t < S_t \leq \theta_t + \delta$	Continuous ≥ 2 time points	Yellow light reminder
2- Moderate abnormality	$\theta_t + \delta < S_t \leq \theta_t + 2\delta$	Continuous ≥ 2 time points	Orange light warning
3- Severe abnormality	$S_t > \theta_t + 2\delta$	Continuous ≥ 2 time points	Red light alarm

In Table II, δ is the threshold increment step, which is used to construct the anomaly level interval in a hierarchical manner; the duration refers to the shortest continuous time required for the anomaly score to meet the conditions, which is used to enhance the stability of AD and avoid false alarms.

IV. EXPERIMENTAL VERIFICATION IN SUPPLY CHAIN SCENARIOS

A. Experimental Setup

To verify the effectiveness of the Autoformer-based SCLs demand modeling and AW method proposed in this paper, this paper selects real operational data from a large port logistics company between 2020 and 2024 to construct an experimental dataset. Its structure is displayed in Table III.

TABLE III
DATASET STRUCTURE

Data type	Feature Name	Time granularity	Total amount of data	Data sources
Historical order data	Daily order quantity	Day	1827	Order management system
Inventory data	Current inventory level	Day	1827	Resource planning system
Shipping time	Average shipping time	Hour	43848	Port dispatch platform
Market environment	CPI	Month/Day	48	National Bureau of Statistics, State Council holiday release
	PMI		48	
	Holiday signs		1827	

To ensure the timeliness of the evaluation and avoid data leakage, this paper divides the dataset into three independent time periods in chronological order: 1) Training set: January 1, 2020 - December 31, 2022 (1096 days in total); 2) Validation set: January 1, 2023 - December 31, 2023 (365 days in total); 3) Test set: January 1, 2024 - December 31, 2024 (366 days in total). For handling missing values, linear interpolation is used for continuous variables, and forward imputation is used for discrete variables. In time alignment, the order date is used as the primary index, and multi-source data is aligned through outer joins to ensure one record per day. Z-score normalization is applied to continuous features, with the mean and standard deviation calculated based on the training set. In sequence construction, a sliding window method is used to construct input-output pairs, with a window length 30 and a prediction step size 7. Finally, weekday, month, and holiday markers are added, and sine-cosine positional encoding is performed. To establish a reliable benchmark for anomaly detection and evaluation, a rule-based labeling process is adopted, defining supply chain logistics anomalies as "significant deviations from normal operating patterns in demand or operations caused by external disturbances or internal failures." Anomaly identification employs a multi-level set of rules for screening: demand surges (marked as candidate anomalies if the average daily order volume fluctuates by more than ±30% of the past 30-day rolling average for two consecutive days), transportation delays (marked as candidate anomalies if the average transportation time exceeds two standard deviations of the historical average for the same period and lasts for at least 24 hours), and inventory backlogs (marked as candidate anomalies if the inventory turnover rate is below the 10th percentile of the historical average for three consecutive days). The results were then independently reviewed and confirmed by three experts in the field of port logistics. The Fleiss' Kappa coefficient of the review results was 0.76, indicating that the labeling has good reliability. The model parameter settings for this paper are demonstrated in Table IV.

TABLE IV
MODEL PARAMETER SETTINGS

Module	Parameter	Specifications
Input/output	Sliding window length	30
	Prediction step size	7
	Number of fused features	8
	Dimension of target prediction value	1
Autoformer basic parameters	Encoder layers	3
	Decoder layers	2
	Attention head count	8
	Hidden layer dimension	512
Adaptive decomposition parameters	Trend extraction window	25
Trend extraction window	Multi-scale sliding window	[90, 30, 7]
Training parameters	Learning rate	1.00E-04
	Optimizer	Adam
	Dropout ratio	0.1
	Batch size	64
	Max epochs	100

The model was implemented using PyTorch 1.13.1, with Adam as the optimizer and a learning rate of 1×10^{-4}. During training, a model checkpoint (including weights, optimizer state, and the current epoch) was saved every 10 epochs. The final model was selected based on the checkpoint with the lowest WAPE on the validation set. Training logs (loss and evaluation metrics) were saved as a comma separated values (CSV) file for subsequent analysis. To ensure reproducibility, all randomization processes used a fixed seed of 2024.

To comprehensively evaluate the effectiveness of the method, it selects three representative time series forecasting and AD models as baselines for comparison:

1) ARIMA: A classic linear model used for analyzing stationary time series.

2) LSTM: A recurrent neural network with long-term memory, used for modeling nonlinear time series;

3) Informer: A Transformer variant based on sparse attention, suitable for long time series forecasting.

Taking ARIMA, LSTM and Informer as control models, the effectiveness of this method in SCLs demand modeling and AW is evaluated from the two levels of LD prediction accuracy and AW effectiveness.

B. Supply Chain Demand Forecasting Performance

SC demand forecasting emphasizes balancing a model's capacity to predict long-term trends with its responsiveness to local disturbances. To further analyze the model's performance in terms of forecast accuracy, this paper compares the WAPE and RMSE performance of various models at different forecast step sizes:

$$WAPE=\frac{\sum_{t=1}^{n}|y_t-\hat{y}_t|}{\sum_{t=1}^{n}|y_t|} \tag{32}$$

$$RMSE=\sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t-\hat{y}_t)^2} \tag{33}$$

Fig. 3 illustrates the outcomes.

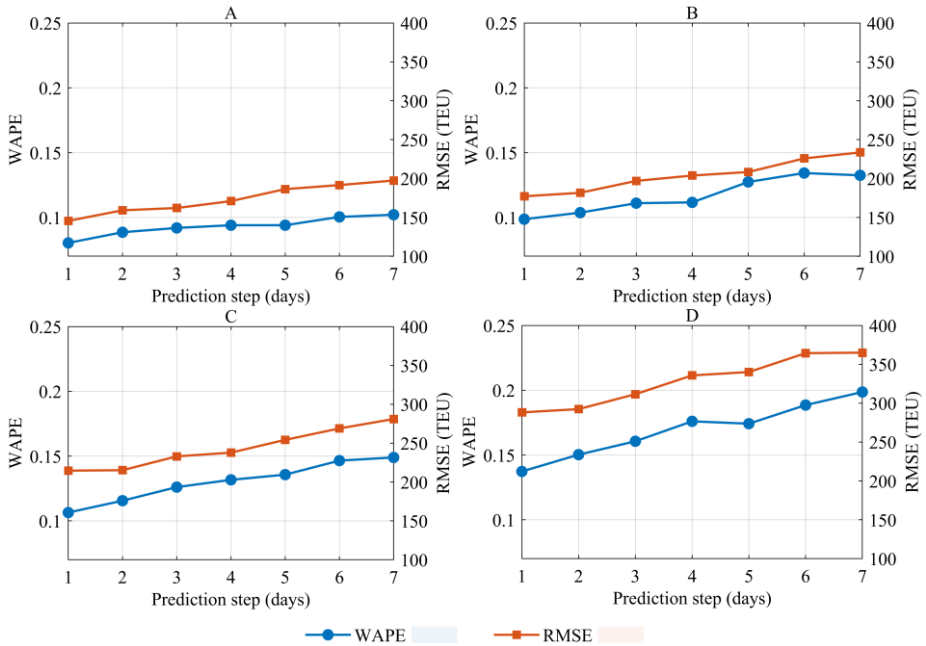


Fig. 3. WAPE and RMSE Comparison Results; A) shows the WAPE and RMSE of the Autoformer, B) shows the WAPE and RMSE of the Informer, C) shows the WAPE and RMSE of the LSTM, D) shows the WAPE and RMSE of the ARIMA.

The outcomes in Fig. 3 demonstrate that the proposed Autoformer-based SCLs demand modeling method outperforms the comparison methods in both WAPE and RMSE metrics. The trends in WAPE and RMSE over the forecast step size indicate that the forecast errors of all methods increase with increasing forecast step size, while Autoformer maintains the lowest forecast error over the 7-day forecast period. In Fig. 3A, its WAPE and RMSE on the seventh day are 0.105 and 200 TEU, respectively. The Autoformer's adaptive trend decomposition mechanism effectively separates long-term trends from short-term fluctuations in LD. Extracting cyclical features in parallel at multiple time scales, such as daily, monthly, and quarterly, fully exploits the multi-level temporal dependencies of PLD, resulting in stronger long-term forecast accuracy. However, the fixed time scale modeling method adopted by the Informer in Fig. 3B is difficult to adapt to the joint impact of multi-level periodic characteristics in PLD; although the LSTM in Fig. 3C has memory capabilities, it is prone to falling into local optimality and cannot effectively handle multi-scale time dependencies; The ARIMA model in Fig. 3D relies on linear assumptions, making it challenging to capture the nonlinear complex relationships and dynamic dependencies among multiple variables in PLD.

To further verify the performance advantage of this method in capturing the changing trend of SCLs demand, it applies a trend matching index. It uses the Spearman rank correlation coefficient within a sliding window to compute the consistency between the predicted and actual sequence in the direction of change. Then it compares and analyzes the performance of different methods in the trend matching index, as shown in Fig. 4.

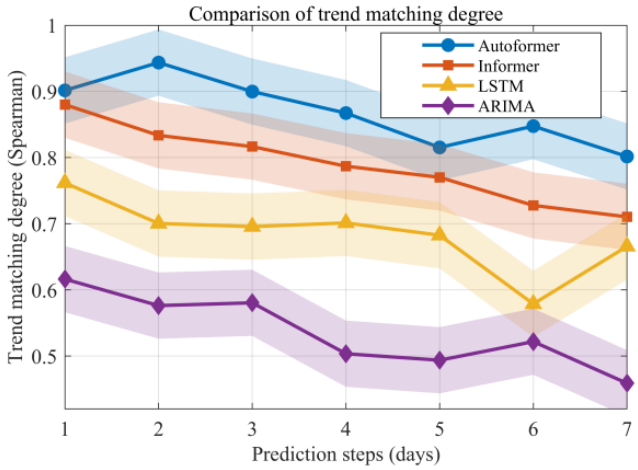


Fig. 4. Trend Matching Comparison Results

As shown in Fig. 4, the proposed Autoformer method generally achieves higher trend matching across different forecast steps. The correlation coefficient shows that the Autoformer achieves a peak trend matching score exceeding 0.900 within a seven-day forecast period, reaching 0.808 on the seventh day. Even with longer forecast steps, the Autoformer maintains high trend consistency. In contrast, the trend matching scores of the Informer, LSTM, and ARIMA methods decrease from initial values of 0.883, 0.768, and 0.615 to 0.712, 0.663, and 0.450, respectively, demonstrating their limitations in long-term trend forecasting. Based on multi-scale modeling and adaptive decomposition, this method decomposes the LD series into long-term trend terms and short-term disturbance terms, accurately modeling each frequency component separately. This effectively avoids trend ambiguity when processing complex time series. Compared to other methods, the Autoformer demonstrates superior trend matching in the complex and dynamic environment of PLD.

C. Verification of the Effectiveness of Supply Chain Anomaly Warning

To assess how well the AW mechanism performs in SCLs scenarios, dynamic thresholds and multi-level judgment rules are set to statistically analyze the classification performance of each model at different anomaly levels. Three core indicators, precision, recall, and F1-score, were used for the quantitative evaluation. Table V displays the results.

TABLE V
COMPARISON OF PRECISION, RECALL, AND F1-SCORE

Model	Abnormal level	Precision	Recall	F1-score
Autoformer	Mild abnormality	0.882	0.857	0.869
	Moderate abnormality	0.835	0.901	0.867
	Severe abnormality	0.916	0.934	0.925
Informer	Mild abnormality	0.765	0.723	0.743
	Moderate abnormality	0.708	0.816	0.758
	Severe abnormality	0.852	0.829	0.840
LSTM	Mild abnormality	0.684	0.652	0.668
	Moderate abnormality	0.621	0.703	0.659
	Severe abnormality	0.789	0.751	0.770
ARIMA	Mild abnormality	0.543	0.587	0.564
	Moderate abnormality	0.502	0.624	0.556
	Severe abnormality	0.658	0.603	0.629

As shown in Table V, the proposed method obtains F1-scores of 0.869, 0.867, and 0.925, respectively. The proposed method performs particularly well in severe AD, with F1-score improvements of 10.1%, 20.1%, and 47.1% over the Informer, LSTM, and ARIMA methods, respectively. The traditional ARIMA method exhibits the lowest detection performance at all anomaly levels, primarily due to its linear modeling assumptions, which are hard to adapt to the complex nonlinear characteristics of port logistics data. While the LSTM method accounts for temporal dependencies, it has limitations when handling multivariate coordinated anomalies. While the Informer improves modeling capabilities through its attention mechanism, its ability to identify anomalies of varying severity is relatively limited.

To directly reflect the model's effectiveness in detecting and warning real risks, it uses FN to quantify the underreporting of each model at different anomaly levels. It selects supplier anomalies, inventory anomalies, logistics node anomalies, and market fluctuation anomalies from port logistics in 2023 as the test set. Fig. 5 demonstrated the outcomes.

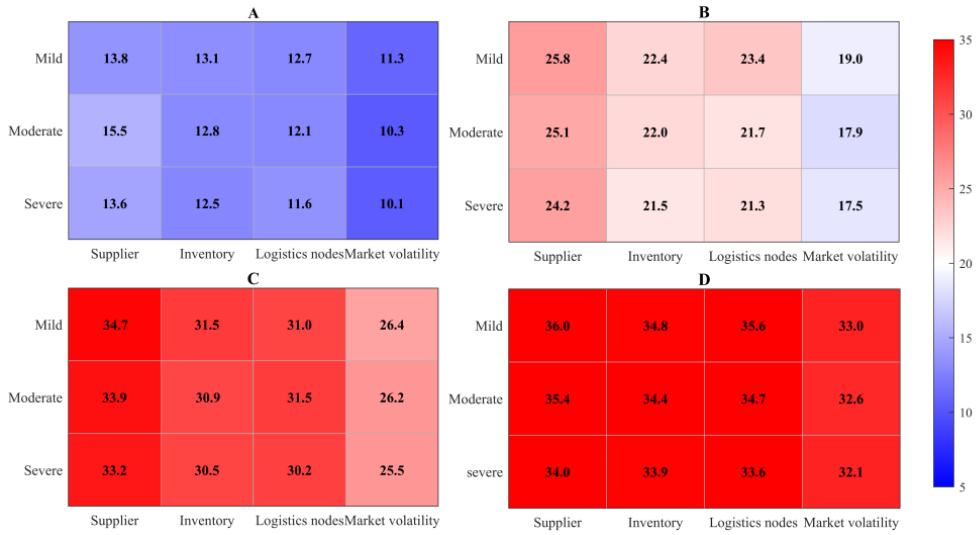


Fig. 5. FNR Comparison Results; A) shows the FNR of the Autoformer, B) shows the FNR of the Informer, C) shows the FNR of the LSTM, D) shows the FNR of the ARIMA

The results in Fig. 5 demonstrate that the Autoformer demonstrates significant advantages in detecting all four anomaly types. In the four severe anomaly scenarios, the average FNR of the Autoformer in Fig 5A is 12.0%, which is 9.1% lower than that of the Informer in Fig 5B (21.1%), 17.9% lower than that of the LSTM in Fig 5C (29.9%), and 21.4% lower than that of the ARIMA in Fig 5D (33.4%). This demonstrates that the Autoformer's adaptive trend decomposition effectively captures complex anomalies. The MSAM allows the model to detect anomaly features across various time scales, improving its capacity to recognize complex anomaly patterns. A dynamic thresholding mechanism based on prediction residuals avoids the limitations of traditional fixed threshold methods and enables adaptive detection of different types of anomalies. This synergistic effect makes the method more reliable in dealing with the complex and ever-changing anomaly scenarios in port logistics systems.

Further, the FPR is calculated to compare the proportion of standard data samples that are incorrectly identified as anomalies by different AD models to measure the specificity of the model in logistics early warning. Fig. 6 illustrates the findings.

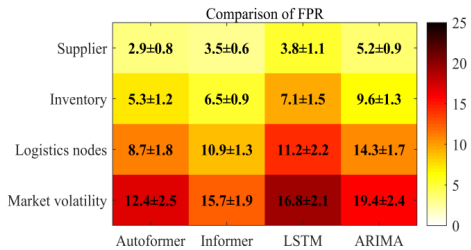


Fig. 6. FPR Comparison Results

As displayed in Fig. 6, for the four warning scenarios, the model in this paper achieves an average FPR of 7.3%; the Informer, LSTM, and ARIMA models achieve 9.2%, 9.7%, and 12.1%, respectively. Comparative results show that the Autoformer model outperforms the Informer, LSTM, and ARIMA models. In the four scenarios of SC, inventory, logistics nodes, and market fluctuations, short-term sharp fluctuations and cyclical structural misalignments can easily lead conventional models to misclassify them as anomalies. However, the model, through scale fusion and residual modeling, proactively accounts for these variations, significantly reducing the false alarm rate. Applying trend-disturbance decoupling at the encoding stage prevents short-term fluctuations from interfering with long-term trend modeling. The MSAM further improves the model's accuracy in discerning anomaly boundaries at different time series frequencies, resulting in more favorable FPR results and effectively improving warning performance.

Further, the average detection delay of each model was compared, and the time difference from the appearance of abnormal features in the input data to the output of the warning signal by the model was recorded. The findings of each model's statistical analysis, based on the backtest of abnormal event samples, are presented in Table VI.

TABLE VI
AVERAGE DETECTION LATENCY COMPARISON

Model	Mild abnormality (h)	Moderate abnormality (h)	Severe abnormality (h)	Overall average (h)
Autoformer	1.8±0.3	1.2±0.2	0.9±0.2	1.3±0.4
Informer	3.2±0.6	2.7±0.5	2.1±0.4	2.7±0.8
LSTM	4.1±0.9	3.5±0.7	2.8±0.6	3.5±1.1
ARIMA	6.5±1.5	5.3±1.2	4.2±1.0	5.3±1.6

Table VI shows that the four models exhibit varying detection latency: Autoformer has the lowest overall average latency, reaching 1.3 ± 0.4 hours, while ARIMA has the highest, reaching 5.3 ± 1.6 hours. Informer and LSTM have average overall latencies of 2.7 ± 0.8 hours and 3.5 ± 1.1 hours, respectively. The comparative results demonstrate that Autoformer's adaptive decomposition and residual-driven early warning mechanism enable it to maintain a relatively fast response in AD.

To further validate the model's practicality and reliability in logistics anomaly early warning tasks, a weighted Kappa coefficient is used to measure the consistency between predictions and true levels. Key performance indicators, such as the over-threshold deviation rate and severe anomaly recognition rate, are also statistically analyzed, as displayed in Fig. 7.

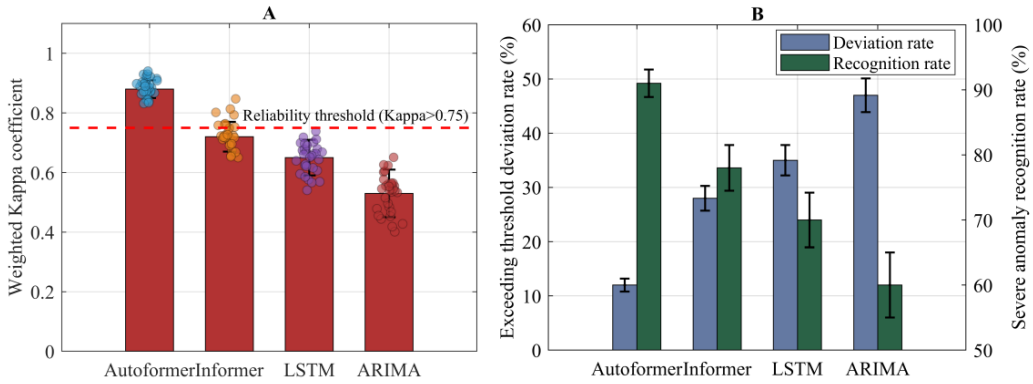


Fig. 7. Comparison of Abnormal Warning Level Consistency; A) shows the weighted Kappa coefficient, B) shows the key performance indicators

As shown in Fig. 7, the Autoformer demonstrates superior performance across all metrics. The weighted Kappa coefficient in Fig. 7A is as high as 0.844, significantly exceeding the reliability threshold of 0.75, indicating a high degree of consistency between its predicted grades and the ground truth. The Informer performs second (Kappa = 0.728), while the Kappa coefficients for the LSTM and ARIMA models are 0.650 and 0.537, respectively. Among the key performance indicators in Fig. 7B, the Autoformer achieves the lowest over-threshold deviation rate (12%) and a severe anomaly recognition rate exceeding 90%, demonstrating not only overall accurate predictions but also effective warning of critical, high-risk events in SCLs. In contrast, the ARIMA model exhibits the weakest performance, with a deviation rate of 47% and a recognition rate of only 60%. While the LSTM model demonstrates some recognition capability, it suffers from significant deviation. The Informer performs moderately well in complex models, outperforming the LSTM and ARIMA models but not the Autoformer. Overall, the Autoformer demonstrates superior performance in terms of grade prediction accuracy, stability, and key event capture, making it suitable for AW in highly dynamic and volatile logistics environments.

D. Ablation Experiment

To verify the contribution of each module to the model performance, an ablation experiment was designed. Related components were removed or replaced sequentially, and their performance was evaluated under the same experimental settings. Six model variants were constructed for comparison: 1) Complete model (Proposed); 2) Quarterly branch removed: only daily and monthly scale attention branches were retained; 3) Monthly branch removed: only daily and quarterly scale attention branches were retained; 4) Dual-path modeling removed: the original Autoformer fixed decomposition method was used instead of the adaptive trend-perturbation dual-path structure; 5) Static Z-score thresholding: the dynamic residual scoring and multi-level thresholding were replaced with the traditional static Z-score method; 6) Single path + static thresholding only: both monthly and quarterly branches in the multi-scale attention were removed, and a static thresholding method was used. The performance comparison of each variant on the two tasks of demand forecasting and anomaly warning is shown in Table VII.

TABLE VII
ABLATION TEST RESULTS

MODEL	WAPE	RMSE (TEU)	F1 (SERIOUS ABNORMALITY)	FNR (%)	FPR (%)	KAPPA
FULL MODEL (PROPOSED)	0.105	200	0.925	12	7.3	0.844
REMOVE QUARTERLY BRANCHES	0.118	225	0.891	15.2	8.1	0.812
REMOVE MONTHLY BRANCHES	0.121	231	0.885	16	8.5	0.806
FIXED DECOMPOSITION	0.129	245	0.872	18.7	9	0.781
STATIC THRESHOLD	0.107	203	0.868	17.5	10.4	0.765
SINGLE PATH + STATIC THRESHOLD	0.135	252	0.821	22.3	11.8	0.732

As shown in Table VII, removing both quarterly and monthly branches led to an increase in WAPE and RMSE, and a decrease in F1, indicating that quarterly and monthly scales are crucial for capturing medium- to long-term cyclical fluctuations. Fixed

decomposition significantly outperformed the full model across all metrics, demonstrating that adaptive decomposition more effectively decouples trends from disturbances and improves the model's ability to model non-stationary sequences. While static thresholds were close to the full model in prediction error, they performed significantly worse in anomaly detection, indicating that dynamic thresholds and multi-level judgments significantly improve the ability to distinguish between anomalies of different degrees and reduce false alarms. The simplified structure and static thresholds exhibited the worst performance, suggesting a synergistic enhancement effect among the modules.

E. Comparison of SOTA Methods

To comprehensively evaluate the effectiveness of this method, five representative time series forecasting and anomaly detection models were selected for comparison to ensure the comparison's sophistication and fairness. Each model was implemented using its official codebase and employed the same data partitioning, preprocessing procedures, and evaluation protocol as this method.

1) PatchTST is a Transformer architecture based on channel-independent patches, exhibiting excellent performance in long-term prediction tasks.

2) TimesNet transforms time series data into a two-dimensional space to model multi-period features, widely regarded as a general-purpose time series model.

3) iTransformer employs an inverted attention mechanism, achieving significant performance in multivariate time series prediction.

4) Autoformer (original) is a classic time series Transformer model using autocorrelation decomposition, serving as the foundational version of this method.

The anomaly detection performance comparison is shown in Table VIII:

TABLE VIII
PERFORMANCE COMPARISON OF SOTA METHODS FOR ANOMALY DETECTION

MODEL	F1 (SERIOUS ABNORMALITY)	F1 (MINOR ABNORMALITY)	FNR (%)	FPR (%)	KAPPA
PROPOSED	0.925	0.869	12.0	7.3	0.844
PATCHTST	0.891	0.832	15.6	9.1	0.815
TIMESNET	0.902	0.841	14.8	8.7	0.826
ITRANSFORMER	0.898	0.838	15.2	9.0	0.819
ORIGINAL AUTOFORMER	0.867	0.812	18.7	9.5	0.781

As shown in Table VIII, compared with PatchTST, TimesNet, iTransformer, and the original Autoformer, the model in this paper has a greater advantage in anomaly detection performance, with a severe anomaly F1 score of over 0.9, specifically 0.925. Except for TimesNet (0.902), the other models all have scores below 0.9, and their prediction errors are all higher than that of the model in this paper. Overall, the proposed method performs better in terms of prediction accuracy, anomaly detection capability, and early warning consistency, verifying the effectiveness and advancement of multi-scale attention, adaptive decomposition, and dynamic threshold mechanisms in complex time-series scenarios of port logistics.

F. Parameter Sensitivity Analysis

To verify the robustness of the critical design choices, we conducted sensitivity analyses on the sliding window length, multi-scale window length, and threshold coefficient. These are shown in Tables IX, 10, and 11.

TABLE IX
SLIDING WINDOW LENGTH SENSITIVITY ANALYSIS

DAYS	WAPE	RMSE	F1 (SERIOUS ABNORMALITY)	TRAINING TIME (EPOCH/MIN)
15	0.121	228	0.901	3.2
30	0.105	200	0.925	4.5
45	0.108	205	0.918	6.1
60	0.113	212	0.91	8.3

As shown in Table IX, when the sliding window length is 30, the model achieves near-optimal performance in both WAPE and F1 scores. Too short a window (15 days) leads to a decrease in predictive performance due to insufficient historical information, while too long a window (60 days) introduces excessive noise and significantly increases computational costs.

TABLE X
MULTISCALE WINDOW LENGTH SENSITIVITY ANALYSIS

COMBINATION	TREND MATCHING DEGREE	F1 (SERIOUS ABNORMALITY)	KAPPA
(20, 60)	0.861	0.903	0.811
(30, 90)	0.9	0.925	0.844
(40, 120)	0.878	0.915	0.829

When the monthly window is 30 and the quarterly window is 90, which aligns with the natural business cycle, the model achieves optimal performance in trend matching, severe anomaly detection (F1), and Kappa. This indicates that defining multi-scale windows based on natural cycles can more effectively extract time-series patterns that correspond to the decision-making level of logistics management. Windows that are too long or too short will lead to feature extraction bias and performance degradation.

TABLE XI
SENSITIVITY ANALYSIS OF DYNAMIC THRESHOLD COEFFICIENT

THRESHOLD	F1 (SERIOUS ABNORMALITY)	FNR (%)	FPR (%)
1.5	0.938	9.2	12.6
2	0.925	12	7.3
2.5	0.901	16.5	5.1
3	0.872	21.3	3.8

When the threshold is 2.0, the model achieves the best balance between FPR and FNR, and has the highest overall score. Lowering the threshold to 1.5, while capturing more anomalies, significantly increases false alarms and reduces warning specificity; raising the threshold above 2.5 leads to a substantial increase in missed alarms. This verifies that choosing a threshold of 2.0 can maintain overall stability while ensuring warning sensitivity.

G. Method Generalization Ability Analysis

To explore the applicability of the proposed method in relevant scenarios, supplementary experiments were conducted on a publicly available supply chain dataset. The model structure remained unchanged; only the input dimensions were adjusted based on the data features, and the model was retrained. The results are shown in Table XII.

TABLE XII
GENERALIZATION ABILITY ANALYSIS

MODEL	WAPE	RMSE	F1 (SERIOUS ABNORMALITY)	TRAINING TIME (EPOCH/MIN)
PATCHTST	0.183	415	0.812	5.1
TIMESNET	0.176	398	0.826	6.3
PROPOSED	0.162	372	0.845	7.2

Table XII shows that the proposed method maintains relatively ideal overall performance without requiring specific structural adjustments to the dataset. This initially indicates that the proposed multi-scale fusion, adaptive decomposition, and dynamic early warning mechanism has a certain degree of generality for logistics time-series data with similar periodicity and non-stationarity. However, its effectiveness in scenarios with significant differences still needs further verification. Future work should further develop a lighter and more adaptive architecture to improve cross-domain generalization capabilities.

V. CONCLUSION

Taking into full account the support capabilities of modern information and communication infrastructure for real-time intelligent decision-making, this paper proposes a deep time series modeling framework for cyber-physical fusion supply chain systems. This paper addresses the unique complexity of port logistics time-series data by systematically integrating and improving self-transformation decomposition, multi-scale attention, and dynamic residual scoring mechanisms through scenario-oriented architecture integration and optimization, in order to decouple trends and disturbances in cargo throughput. Furthermore, it constructs an anomaly scoring function using a residual-driven dynamic threshold mechanism to improve the sensitivity and robustness of the warning system. Experimental results show that, within a 7-day forecast period, the proposed model achieves relatively lower WAPE and RMSE in cargo throughput forecasting compared to the baseline. Its performance on trend matching indicators demonstrates its good trend-following ability. Through multi-scale modeling, trend construction, and anomaly identification, this paper provides a certain reference for intelligent risk perception and response in high-volatility port logistics scenarios.

COMPETING INTERESTS

No competing interests are disclosed by the authors.

AUTHORSHIP CONTRIBUTION STATEMENT

Xizhuo Han: Supervision, Conceptualization, Project administration, Writing-Original draft preparation.

Chuanchuan Teng: Validation, Methodology, Software.

DATA AVAILABILITY

Available upon request.

DECLARATIONS

Not applicable.

CONFLICTS OF INTEREST

The writers assert that they have no conflict of interest pertaining to the release of this article.

AUTHOR STATEMENT

All authors have reviewed and endorsed the manuscript, confirming it fulfills the previously mentioned criteria for authorship. Furthermore, each author is confident that the manuscript accurately reflects genuine research.

FUNDING

Not applicable.

ETHICAL APPROVAL

Each author was directly and significantly involved in the work that resulted in this paper, and they will all accept public accountability for what it contains.

REFERENCES

[1] A. Aamer, L. Eka Yani, and Im. Alan Priyatna, "Data analytics in the supply chain management: Review of machine learning applications in demand forecasting," *Operations and Supply Chain Management: An International Journal*, vol. 14, no. 1, pp. 1–13, 2020.

[2] K. Douaioui, R. Oucheikh, O. Benmoussa, and C. Mabrouki, "Machine Learning and Deep Learning Models for Demand Forecasting in Supply Chain Management: A Critical Review.," *Applied System Innovation (ASI)*, vol. 7, no. 5, 2024. DOI: 10.3390/asi7050093

[3] M. R. Bhuiyan, S. I. Rohan, F. Rahman Sheikh, and M. S. Alam, "Supply Chain Risk Management: Strategic Solutions for Reducing Transportation and Logistics Risks," *Academic Journal of Innovation, Engineering & Emerging Technology*, vol. 1, no. 1, pp. 72–90, 2024. DOI: 10.69593/ajieet.v1i01.125

[4] H. Panjehfouladgaran and S. F. W. T. Lim, "Reverse logistics risk management: identification, clustering and risk mitigation strategies," *Management Decision*, vol. 58, no. 7, pp. 1449–1474, 2020. DOI: 10.1108/MD-01-2018-0010

[5] K. Lai, M. Vejvar, and V. Y. H. Lun, "Risk in port logistics: Risk classification and mitigation framework," *International Journal of Shipping and Transport Logistics*, vol. 12, no. 6, pp. 576–596, 2020. DOI: 10.1504/IJSTL.2020.111117

[6] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Adv Neural Inf Process Syst*, vol. 34, pp. 22 419–22 430, 2021.

[7] L. Li, Y. Gong, Z. Wang, and S. Liu, "Big data and big disaster: a mechanism of supply chain risk management in global logistics industry," *International Journal of Operations & Production Management*, vol. 43, no. 2, pp. 274–307, 2023. DOI: 10.1108/IJOPM-04-2022-0266

[8] S. Yerra, "Optimizing supply chain efficiency using AI-driven predictive analytics in logistics," *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, vol. 11, no. 2, pp. 1212–1220, 2025. DOI: 10.32628/CSEIT25112475

[9] B. He and L. Yin, "Prediction modelling of cold chain logistics demand based on data mining algorithm," *Math Probl Eng*, vol. 2021, no. 1, p. 342 1478, 2021. DOI: 10.1155/2021/3421478

[10] L. Terrada, M. El Khaili, and H. Ouajji, "Demand forecasting model using deep learning methods for supply chain management 4.0," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 5, 2022.

[11] J. Zhu, Y. Wu, Z. Liu, and C. Costa, "Sustainable optimization in supply chain management using machine learning," *International Journal of Management Science Research*, vol. 8, no. 1, pp. 1–8, 2025.

[12] H. Xu, G. Pang, Y. Wang, and Y. Wang, "Deep isolation forest for anomaly detection," *IEEE Trans Knowl Data Eng*, vol. 35, no. 12, pp. 12 591–12 604, 2023. DOI: 10.1109/TKDE.2023.3270293

[13] Y. Chen, H. Zhang, X. Yan, and Q. Miao, "Supply chain demand forecasting based on multi-time scale data fusion network," *Comput Ind Eng*, p. 111 324, 2025. DOI: 10.1016/j.cie.2025.111324

[14] X. I. E. Luqiang, X. U. Jing, and W. Haiyan, "Risk early warning model of cruise ship construction material logistics collection and distribution based on RS-RBFNN," *China Safety Science Journal*, vol. 33, no. 6, p. 114, 2023.

[15] X. Ma and H. Zhang, "Time Series Forecasting Method Based on Multi- Scale Feature Fusion and Autoformer," *Applied Sciences*, vol. 15, no. 7, p. 3768, 2025. DOI: 10.3390/app15073768

[16] D. Cao and S. Zhang, "AD-autoformer: decomposition transformers with attention distilling for long sequence time-series forecasting," *J Supercomput*, vol. 80, no. 14, pp. 21 128–21 148, 2024. DOI: 10.1007/s11227-024-06266-8

[17] Z. Tao, Q. Xu, X. Liu, and J. Liu, "An integrated approach implementing sliding window and DTW distance for time series forecasting tasks," *Applied Intelligence*, vol. 53, no. 17, pp. 20 614–20 625, 2023. DOI: 10.1007/s10489-023-04590-9

[18] M. A. Jahin, A. Shahriar, and M. Al Amin, "MCDFN: supply chain demand forecasting via an explainable multi-channel data fusion network model," *Evol Intell*, vol. 18, no. 3, p. 66, 2025. DOI: 10.1007/s12065-025-01053-7

[19] W. Zheng *et al.*, "Design of a modified transformer architecture based on relative position coding," *International Journal of Computational Intelligence Systems*, vol. 16, no. 1, p. 168, 2023. DOI: 10.1007/s44196-023-00345-z

[20] N. M. Foumani, C. W. Tan, G. I. Webb, and M. Salehi, "Improving position encoding of transformers for multivariate time series classification," *Data Min Knowl Discov*, vol. 38, no. 1, pp. 22–48, 2024. DOI: 10.48550/arXiv.2305.16642

[21] C. Duan, G. Xiu, and Y. Zhang, "Coordinated management method of information contract in port logistics service supply chain," *J Coast Res,* vol. 93, no. SI, pp. 1047–1052, 2019. DOI: 10.2112/SI93-151.1

[22] L. Shuo, H. Zehua, W. Shuo, Y. Qian, and L. Wenbo, "Research on Enterprise Logistics Management Based on Big Data," in *2024 Second International Conference on Data Science and Information System (ICDSIS)*, IEEE, 2024, pp. 1–4. DOI: 10.1109/ICDSIS61070.2024.10594210

[23] Y. Chen, W. Jia, and Q. Wu, "Fine-scale deep learning model for time series forecasting," *Applied Intelligence*, 2024. DOI: 10.1007/s10489-024-05701-w

[24] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep learning for time series forecasting: a survey," *Big Data,* vol. 9, no. 1, pp. 3–21, 2021. DOI: 10.1089/big.2020.0159

[25] Q. Yu, G. Yang, X. Wang, Y. Shi, Y. Feng, and A. Liu, "A review of time series forecasting and spatio-temporal series forecasting in deep learning," *J Supercomput*, vol. 81, no. 10, pp. 1–48, 2025. DOI: 10.1007/s11227-025-07632-w

[26] G. Tian, C. Zhang, Y. Shi, and X. Li, "MultiWaveNet: A long time series forecasting framework based on multi-scale analysis and multi-channel feature fusion," *Expert Syst Appl*, vol. 251, p. 124 088, 2024. DOI: 10.1016/j.eswa.2024.124088

[27] G. Su and Y. Guan, "MSDformer: an autocorrelation transformer with multiscale decomposition for long-term multivariate time series forecasting," *Applied Intelligence*, vol. 55, no. 3, p. 179, 2025. DOI: 10.1007/s10489-024-06105-6

[28] S. Feng *et al.*, "Multi-scale attention flow for probabilistic time series forecasting," *IEEE Trans Knowl Data Eng*, vol. 36, no. 5, pp. 2056–2068, 2023. DOI: 10.1109/TKDE.2023.3319672

[29] C. Lei, H. Zhang, Z. Wang, and Q. Miao, "Multi-Model Fusion Demand Forecasting Framework Based on Attention Mechanism," *Processes*, vol. 12, no. 11, p. 2612, 2024. DOI: 10.3390/pr12112612

[30] J. Li, L. Zhu, Y. Zhang, D. Guo, and X. Xia, "Attention-based multi-scale prediction network for time-series data," *China Communications*, vol. 19, no. 5, pp. 286–301, 2021. DOI: 10.23919/JCC.2021.00.005

[31] J. Lu, L. Pan, and Q. Ren, "Spatial-temporal memory enhanced multi-level attention network for origin-destination demand prediction," *Complex & Intelligent Systems*, vol. 10, no. 5, pp. 6435–6448, 2024. DOI: 10.1007/s40747-024-01494-0

Chuanchuan Teng graduated from Dalian Maritime University in 2016 with a Master's degree in logistics engineering and management, and now he is working in Cangzhou Jiaotong College. His research interest includes supply chain management and port logisitics.

Xizhuo Han graduated from Yanshan University in 2020 with a Master's degree in logistics engineering, and now she is working in Cangzhou Jiaotong College. Her research interest includes supply chain management and port logisitics.

Federated Reinforcement Learning for Load Balancing in O-RAN 5G and Beyond Networks

Mohammed Imad Aal-Nouman, Sarmad M. Hadi and Haider A. H. Alobaidy

Abstract—**The intelligent distributed control is a critical factor in next-generation mobile networks for connecting a massive number of devices and handling diverse services. The Open Radio Access Network (O-RAN) has recently gained significant traction as a flexible architecture based on standards that enables disaggregated, programmable network components. However, because O-RAN is inherently distributed, it creates an imbalance in traffic load when user mobility and traffic are dynamic. This paper proposes a Federated Reinforcement Learning (FedRL) framework to help load balance O- RAN units (O-RUs). The framework enables each O-RU to train a local reinforcement learning agent using real-time information on signal quality, mobility dynamics, and load conditions. The trained agents periodically share their learned information via federated averaging of their Q-tables at the Near-Real-Time RAN Intelligent Controller (Near- RT RIC), enabling collaborative learning without requiring centralized data collection. Using a synthetic dataset obtained from MATLAB-generated simulations of a mobile network of 500 m x 500 m, the proposed system is trained and tested. It simulates realistic user movement and radio conditions between multiple cells. In the experiments, FedRL reduces load imbalance by over 50% compared to rule-based schemes and by around 25% compared to centralized RL. It also achieves fairness values exceeding 0.95, increases throughput by more than 15%, increases handover success rates by around 13%, and reduces ping-pong events by 30-40 % compared to the baselines.**

Index Terms—**O-RAN, Load balance, B5G, Federated Reinforcement Learning, Handover.**

I. INTRODUCTION

Mobile communication has advanced from the early analog first-generation (1G) systems to the advanced capabilities of fifth-generation (5G), with sixth-generation (6G) on the horizon. These changes have led to significant shifts in how networks are designed and operated [1]. Over time, traditional Radio Access Network (RAN) designs have struggled to keep up with rising demands for flexibility, scalability, and cost efficiency. To overcome these challenges, the Open-RAN (O-RAN) was introduced as a new architectural model. O-RAN focuses on openness, intelligent control, and interoperability [2].

Formed in 2018, the O-RAN Alliance launched the Open RAN initiative as a new approach to rethink traditional RAN designs. The purpose of this is to disassemble the previous 'system structure', which is still held intact by standardized interfaces and open communication protocols [3]. O-RAN, through the separation of hardware and software as well as interoperability of equipment from different vendors, creates an environment for innovation that reduces dependence on a single supplier and enables cheaper network deployment [4]. The Near-Real-Time RAN Intelligent Controller (RT RIC) and Service Management and Orchestration (SMO) platform are key components introduced by this architecture that can improve how the RAN is programmed, managed, and optimized [5]. The O-RAN architecture consists of a radio domain and a management domain architecture. The radio architecture contains four significant elements. The O-RU is responsible for low-level physical (PHY) layer functions and radio frequency (RF) signals. The Open Distributed Unit (O-DU) handles MAC, RLC and High-PHY layer-related tasks. The O-CU-CP handles control signaling, while the O-CU-UP is for user data traffic [6]. In operational terms, the Service Management and Orchestration platform is responsible for controlling and optimizing the network. The SMO's Non-Real-Time RIC (Non-RT RIC) leverages the A1 interface to facilitate long-term policy enforcement and the training of Artificial Intelligence (AI) and Machine Learning (ML) models. Simultaneously, the Near-Real-Time RIC (Near-RT RIC) facilitates faster decision-making by hosting dynamic xApps that interact directly with the O-DU and O-CU for optimization purposes [4].

The O-RAN is driven by several intersecting demands. The growth of mobile data is accelerating rapidly, driven by new applications such as augmented reality, ultra-high-definition video streaming, and the IIoT network. Thus, demand for low latency and intelligent resource allocation is growing [7]. Operators are under mounting pressure to achieve lower operational costs while retaining the same level of quality of service (QoS). Also, the proprietary, closed designs of conventional RAN platforms have historically hampered innovation and limited flexible large-scale deployments [8]. O-RAN tackles these issues by enabling intelligence based on AI and ML through open interfaces while supporting adaptation and optimization at both the local and network-wide level [9]. Despite the architecture, O-RAN's failure to load balance is an issue. When the O-RUs traffic is imbalanced, it can lead to poor user experience, increased latency, and inefficient spectrum use. Most load-balancing techniques leverage either a handover policy based on signal strength or a fixed threshold. These do not respond to any real-time traffic or mobility changes [10-13]. Centralized control approaches are more coordinated, but they are prone to latency and scalability issues in large-scale heterogeneous networks [14]. While recent works explore the use of

Mohammed Imad Aal-Nouman and Haider A. H. Alobaidy are faculty members at the Department of Information and Communications Engineering, College of Information Engineering, AL-Nahrain University, Baghdad, Iraq (e-mail: m.aalnouman@nahrainuniv.edu.iq, and Haideralobaidy@nahrainuniv.edu.iq)

Sarmad M. Hadi is a faculty member at the Department of Computer Networks Engineering, College of Information Engineering, AL-Nahrain University, Baghdad, Iraq (e-mail: sarmad@nahrainuniv.edu.iq)

DOI: w

reinforcement learning and federated learning for resource optimization in O-RAN, most such approaches remain limited. Centralized RL techniques face limitations in latency and scalability in large deployments [15], while multi-agent RL solutions allow local adaptation but typically lack coordination, leading to inconsistent results [16]. Existing federated learning approaches have focused on slice-level orchestration or static allocation [17-19], but not on those that account for mobility-driven handover events. More recent papers also point out that the lack of mobility adaptation ultimately limits the efficiency of FL-based schemes under realistic O-RAN conditions [20, 21]. The identified gaps demonstrate the need for a mobility-aware federated load-balancing mechanism that guarantees scalability, privacy and responsiveness in a real-world O-RAN scenario. This paper analyzes the dynamic and intelligent load balancing challenge in an O-RAN-based 5G Network and beyond.

Federated Reinforcement Learning (FedRL) framework offers a formulation in which distributed agents associated with an O-RU make local decisions on handovers and load distribution. These decisions depend on the current states within the network. Federated learning has agents synchronizing their models periodically using the Near-RT RIC, allowing learning without exchanging data. The purpose of this research is to enhance load balancing performance while ensuring scalability, adaptability, and user privacy. We use a synthetic mobile network scenario to generate a detailed dataset. Each record in the dataset contains information on user mobility, radio signal quality, and local cell load. FedRL is assessed using basic metrics such as Max-to-Min Load Difference, Handover Failure Rate, and Ping-Pong Occurrences, compared with centralized and multi-agent RL. Most centralized RL approaches assume global state observability (i.e., all agents are fully observable) and that centralized control involves only a single-point control decision. Also, classic multi-agent RL schemes operate independently without coordination. However, the proposed FedRL-O-RAN framework proposes coordinated yet decentralized learning for mobility-aware handover optimization. The main contribution of this work is threefold. First, rather than directly selecting the target cell to handover, the approach formulates the handover control as A3-offset learning. Second, the basic building block of this approach is tabular Q-learning implemented at each O-RU to keep computation simple and easy to understand. Finally, the learning mechanism employs federated Q-table aggregation at the Near-RT RIC to achieve consistency in their handover policies without disclosing any user data. This combination enables scalable, privacy-preserving, mobility-aware load balancing within the operational constraints of O-RAN. The principal contributions of this paper are summarized as follows:

- Development of distributed Q-learning agents, one per O-RU, synchronized by means of federated averaging through the Near-RT RIC.
- Inclusion of signal quality, cell load, and user mobility context in the state inputs that make the decision.
- The development of the MATLAB multi-cell simulation framework, which includes interference, mobility, and dynamic traffic and is evaluated against rule-based, centralized RL, and MARL baselines.

The rest of the paper is organized as follows. Related work is discussed in Section II; Section III presents the system model architecture and methodology; Section IV presents all the simulation parameters and metrics; Section V highlights the main findings and results; and finally, conclusions are drawn in Section VI.

II. RELATED WORKS

Intelligent load balancing and handover optimization are playing a central role in O-RAN and C-RAN performance, especially with the integration of reinforcement learning and federated learning. As network densification increases and mobility becomes more dynamic, intelligent decision-making mechanisms are increasingly required to maintain fairness, throughput stability, and handover reliability.

From a centralized optimization perspective, Pamuklu et al. (2021) present an RL-based strategy for adaptively managing function splitting between CU and DU components in disaggregated O-RAN settings, aiming to minimize operational costs in energy-constrained scenarios [23]. Ali (2025) showed that when using centralized learning strategies in C RAN environments, it improves energy efficiency and spectral efficiency and reduces BBU centralized deployments' handover failure rate [24]. Researchers have studied DDPG-based reinforcement learning techniques for dynamic optimization of handover parameters in dense 5G scenarios, which can minimize ping-pong effect and handover failure rate [25]. Gupta and others in (2021) implemented joint load balancing and handover optimization with deep RL in a multi-band LTE/5G network. They showed improved stability and throughput for the network [26]. Centralized methods make use of global observability and coordinated decision making, which allows optimization objectives for the whole system to be included in learning directly. Nevertheless, difficulties when it comes to scaling and signaling could arise for these techniques in large O-RANs, especially when rapid mobility dynamics require low-latency local adaptation.

Orhan et al. (2021) presented a Graph Neural Network-based RL xApp for intelligent user association of the O RAN RIC to move towards distributed and multi-agent learning approaches. The RL xApp from Orhan et al. manages to achieve 20 to 45% gains in load balancing in comparison to greedy baselines [15]. In the context of O-RAN, Lai et al. (2023) designed mmLBRA, a multi-agent, multi-armed bandit-based scheme which effectively balances load among O-RUs [16]. Several additional works [16, 18, 26, 27] propose multi-agent systems or hierarchical FL structures for distributing decision-making in O-RAN. When each entity learns from its local environment, then the overall entity can respond to heterogeneous traffic and centralized bottlenecks can be reduced. However, many do not perform a structured policy synchronization across their O-RUs, or rely on centralized inference components. This may detract from achieving consistent global fairness, and lead to unstable performance as traffic patterns change rapidly across neighboring cells.

In O-RAN, collaborative optimization that protects privacy

may be possible with federated learning. Zhang et al. (2022) designed algorithms for slice-aware xApps coordination based on federated deep reinforcement learning (FedDRL) [17]. A federated deep Q-learning xApp for 5G load balancing was developed by Lin et al (2024), which improves throughput and protects privacy [28]. The authors of reference [29], Cao et al., designed a federated DQN access control mechanism in O RAN by using UCB-based UE selection to maximize long-term throughput and minimize unnecessary handovers. Ahmed and colleagues developed a framework that utilizes federated DRL for distributed slice management to optimize O-RAN environments. Zhao et al. also proposed a multi-layer collaborative federated learning (MLCFL) scenario framework for next-generation (6G) O-RAN coordination. Extensions of this line of research include Elastic Federated Learning (EFL) which emphasizes federated RL for spectrum sharing in NTN deployments. The these work show that a federated coordination can trade-off between distributing the intelligence and increasing the performance of the system globally without interfering with the data locality. Nonetheless, several federated methods are primarily tailored for slice management, access control, or spectrum allocation, as opposed to fine-grained, mobility-aware, user-level handover optimization under dynamic load fluctuations.

Frameworks that are hybrid and predictive have also been studied to improve adaptability. In 2023, Kavehmadavani et al. proposed a traffic steering solution using LSTM for edge-centric O-RAN deployments [33] that will help in load steering based on traffic prediction.

Singh et al. (2023) introduces Cascade RL (CaRL) that leverages state-space decomposition across O DU layers to balance QoS trade-offs [34]. Semiari et al. (2025) proposed a graph-RL load balancing mechanism for QoS-guaranteed O RAN slices, allowing achieving consistent resource fairness under diverse traffic mixes [35]. These approaches may enhance both orchestration and predictive capabilities, but they often abstract away from direct control of mobility-driven handover parameters at the O-RU level. Instead, they focus on other coordination objectives at higher layers.

While RL and FL have been used in the past for load balancing (LB) and handover (HO) optimization in O-RAN and C-RAN, a number of papers focus on global coordination, slice level allocation, static functional distribution etc. A constant limitation across these works is the limited introduction of real-time user-level mobility adaptation to federated or distributed decision loops. A deployment with dynamic conditions in the speed of users, signal value, localized congestion and the like can lead to unfairness or ping pong occurrences if a mobility-aware handover control is not used.

Pamuklu et al. [23], for instance, work on dynamic function splitting without user mobility pattern or federated coordination. The authors Orhan et al [15] improve load fairness through RL-driven user association but do not consider explicit model-specific handover design nor distributed model synchronization. Lin and coworkers [28] propose a federated DQN framework. Their experiments, however, do not include dynamic handover logic nor do they use aggressive load conditions. Likewise, Zhang et al. [17] allocate resources across network slices, but not with any granular mobility-aware handover decision-making, while Cao et al. [29] access control, not adaptive handover parameter optimization.

FedRL-O-RAN is a decentralized and user-centric reinforcement learning framework for dynamic handover optimization unlike the previous work. Every O-RU integrates a distributed Q-learning agent that leverages radio signal readings, user mobility patterns, and local cell load to formulate localized decisions. Through the near-RT RIC, periodic model synchronization occurs through federated averaging, which aligns all policies in the system without raw user data sharing. The proposed framework tackles a key limitation of previous works [15, 17, 28] by addressing the challenges of centralized global optimization and fully independent multi-agent control. It explicitly integrates mobility context into distributed learning with federated synchronization.

III. METHODOLOGY

Overall, although prior works have made contributions through architectural novelties or specialized learning schemes, FedRL-O-RAN provides a more holistic, scalable and privacy-aware solution. This study directly integrates user mobility into federated coordination in a realistic O-RAN deployment which bridges important research gaps and offers a practical approach for intelligent handover management and load balancing in 5G and beyond the generations of networks.

The issue of load balancing and handover control in O-RAN was formulated as a Markov Decision Process (MDP), with each Open Radio Unit (O-RU) acting as an independent reinforcement learning agent. Rather than directly choosing a target cell, each agent learns to modify the A3 handover offset. In this way, it is able to control handover aggressiveness, affecting load distribution.

In each of O-RU i's decision steps, O-RU i observes a locally constructed state that incorporates its own current load, the average SINR of its connected users and their average mobility speed. Consequently, the state of O-RU i is defined as:

$$s_i(t) = \left[\rho_i(t), \overline{\mathrm{SINR}}_i(t), \bar{v}_i(t)\right] \quad (1)$$

Where $\rho_i(t)$ denotes the current load of O-RU i, $\overline{\mathrm{SINR}}_i(t)$ represents the average SINR of the UEs served by O-RU i, and $\bar{v}_i(t)$ denotes the average mobility speed of its connected users. Together, these variables describe the load and mobility conditions within the cell. Since all state information is derived from locally available measurements, the representation remains compatible with the distributed nature of O-RAN operation.

The continuous state variables were discretized before learning because tabular Q-learning requires a finite state space. The local load, average SINR, and average mobility speed were mapped into discrete state categories, enabling the agent to maintain a Q-table while preserving the information required for handover decision-making. This representation is compatible with the distributed operation of O-RAN.

From the available action set, the agent selects an action

according to the observed state.

$$a_i(t) \in \{-2,\ 0,\ 2\} \quad (2)$$

The value of the A3 handover offset controls the trigger margin corresponding to these actions, thereby influencing handover decisions. A handover occurs from a serving O-RU to a candidate O-RU only if the RSRP difference between the best candidate and the serving cell exceeds the configured A3 offset. When the offset is adjusted, the learning agent indirectly controls the aggressiveness of handover and load distribution.

After taking an action, the agent receives a reward that reflects the impact of the selected A3 offset on local load conditions and handover stability. To ensure that decisions rely only on local information, the reward is computed at each O-RU using only locally available information rather than network-wide load-balancing metrics. The reward of O-RU i at time t is defined as:

$$r_i(t) = \Delta L_i^{\text{local}}(t) + \beta_1 I_{\text{success}} - \beta_2 I_{\text{failure}} - \beta_3 I_{\text{ping-pong}} \quad (3)$$

where $r_i(t)$ denotes the reward received by O-RU i at time step t, and $\Delta L_i^{\text{local}}(t)$ represents the improvement in the local load condition of O-RU i resulting from the selected action. The binary indicators I_{success}, I_{failure}, and $I_{\text{ping-pong}}$ correspond to successful handovers, failed handovers, and ping-pong handover events, respectively. The weighting coefficients β_1, β_2, and β_3 control the contribution of each component to the overall reward.

This reward formulation guides the agent toward actions that improve local load distribution while maintaining handover stability. The reward increases following successful handovers, while failed handovers and ping-pong events reduce it.

Each O-RU employs a distributed reinforcement learning agent that uses tabular Q-learning to learn state-action optimal mapping. The Q-values are updated at each time step according to the standard temporal-difference update rule:

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right] \quad (4)$$

Where α is the learning rate, γ is the discount factor, r is the observed reward, and $\max_{a'} Q(s',a')$ represents the maximum estimated future value over all possible actions. This update gradually refines the Q-table maintained locally at each O-RU, enabling the agent to learn an effective handover control policy over time.

To enable collaboration without exposing raw user data, the agents engage in a Federated Averaging (FedAvg) process. After a set number of local training episodes, each O-RU sends its local Q-table Q_i to the Near-Real-Time RIC, which computes the global Q-table:

$$Q_{\text{global}} = \frac{1}{N}\sum_{i=1}^{N} Q_i \quad (5)$$

The collected Q-table is then sent to all O-RUs, reinitializing their local agents, so that all O-RUs benefit from each other's experiences. This speeds up learning network. It also aligns your policies and protects users. Thus, FedRL-O-RAN will function as a coordinated/context-aware decision system, which can enable load balancing and stable handovers in dynamic environments. This framework facilitates independent O-RUs' decision-making by leveraging federated learning to jointly enhance the optimization of policies without requiring central data collection. The overall system model is based on the O-RAN architecture, an integration of FedRL to enable intelligence load balancing for 5G and beyond. The scheme comprises multiple O-RUs, a Near-Real-Time RIC, and centralized orchestration by the Service Management and Orchestration layer, as presented in Algorithm 1.

Algorithm 1: Proposed FedRL-O-RAN

Initialize local Q-table Qi for each O-RU$_i$
FOR round = 1 to R:
  FOR each O-RU$_i$ in parallel:
    FOR episode = 1 to E:
      FOR each time t:
        Observe state s_i
        Select action a_t using ε-greedy policy
        Apply $a_i(t)$ as the A3 offset for handover triggering
        Execute association update and observe next state $s_{i(t+1)}$
    Compute local reward $r_i(t)$ using local load improvement, handover outcome, and ping-pong indicator
      Update Q_i using Q-learning Eq(4)
      END FOR
    END FOR
    Upload Q_i to Near-RT RIC
  END FOR
  $Q_{global} = (1/N)\ \Sigma_i\ Q_i$ # FedAvg aggregation
  Broadcast Q_{global} and update $Q_i = Q_{global}$ for all O-RUs
END FOR
  Return $\{Q_i\}$, Q_{global}

In this approach, every O-RU creates a local Q-table capable of selecting an A3 offset. The Near-RT RIC uses FedAvg to aggregate Q-tables and shares global table to synchronize policy without UE data sharing. The simulation unfolds in discrete time steps of 1 second. At every interval, the mobility positions of users are updated first. The local state, si(t), for each O-RU, i, is computed from the aggregated statistics of the UEs served by that O-RU. The A3 offset that is chosen will be activated, while handover decisions will be executed based on the trigger condition. When updates for all associations are completed for the current time step, the loads of all O-RUs are updated, and the reward ri(t) is computed. Subsequently, a local update of the Q-table is performed at each O-RU. It ensures every agent's state changes at the same time.

IV. SIMULATION SETUP

This part discusses the scenario, settings, and reference approaches used to assess FedRL-O-RAN. This paper outlines the simulation environment, provides details of data generation, the performance metrics utilized, and the baseline methods against which the proposed approach is compared.

A. Simulation Environment

The implementation of the suggested FedRL-O-RAN framework and its performance were evaluated in a MATLAB-based simulation environment that emulates the operation of an

O-RAN-enabled mobile network. The simulated network consists of seven O-RUs uniformly and randomly distributed throughout the 500 m × 500 m service area. All O-RUs use the same settings for hardware and power transmission to achieve uniform coverage characteristics. Because this work primarily focuses on modeling load balancing and mobility-aware handover optimization, dynamic power consumption modeling is outside its scope.

The simulation takes place in discrete time with a fixed sampling period of 1 second. At every time step, all O-RUs observe their local state information and simultaneously execute their decision updates according to the synchronization scheme.

Every O-RU acts as an independent reinforcement learning agent utilizing tabular Q-learning. The learning configuration uses a discount factor of $\gamma = 0.95$, a learning rate of $\alpha = 0.001$ and an ε-greedy exploration policy that decays from 1.0 to 0.1.

Through 50 communication rounds, federated learning is executed. After 10 local training episodes, each O-RU sends its local Q-table to the Near-RT RIC each round. To produce a global Q-table, the RIC combines the received Q-tables by using Federated Averaging (FedAvg), which is then sent back to all O-RUs for the next round. Each O-RU will use local experience and knowledge along with the network-wide experience.

All dataset used during training/testing is generated in the simulation environment itself. Every record represents the overall state of an O-RU at a specific timestep, including its load, the average SINR of served users, and the average user mobility speed. The label reflects the chosen A3 offset action. The reward is computed based on load balance improvement and handover outcome. Table 1 presents the features of the state; the definition of actions and the components of the reward.

TABLE 1.
STATE FEATURES, ACTION DEFINITION, AND REWARD COMPONENTS

Category	Name	Description
State Features	Local load $\rho_i(t)$	Number of UEs currently connected to O-RU i
	Average SINR $\overline{\text{SINR}}_i(t)$	Mean SINR of UEs served by O-RU i
	Average speed $\bar{v}_i(t)$	Mean mobility speed of UEs connected to O-RU i
Action (Label)	A3 offset $a_i(t) \in \{-2,0,2\}$	Selected handover trigger margin controlling handover aggressiveness
Reward Components	$\Delta L_i^{\text{local}}(t)$	Local load-improvement term after the selected action
	I_success, I_failure, I_ping-pong	Indicators reflecting successful handover, failed handover, and ping-pong occurrence

The simulation considers a dense small-cell O-RAN deployment covering an area of 500 m × 500 m with seven O-RUs. To evaluate scalability, additional deployment scenarios were examined by varying the coverage area and the number of O-RUs. The objective was to determine whether the proposed FedRL framework can maintain its load-balancing and mobility-management performance as the network size and density increase. The scalability scenarios considered in this study are summarized in Table 2.

Scenario S2 examines the impact of increasing O-RU density within the original deployment area. Scenario S3 evaluates the effect of expanding the coverage area while keeping the number of O-RUs unchanged. Scenario S4 combines both factors.

TABLE 2
SCALABILITY SCENARIOS USED IN THE EVALUATION

Scenario	Area (m × m)	O-RUs
S1	500 × 500	7
S2	500 × 500	14
S3	1000 × 1000	7
S4	1000 × 1000	14

B. Learning Algorithms

In order to assess the performance of the proposed FedRL-O-RAN framework, performance comparison is conducted against three representative baseline methods, which correspond to a different philosophy for load balancing and handover decision of O-RAN networks. These baselines incorporate both learning-free heuristics and learning-based techniques for a fair and comprehensive evaluation.

Rule-Based Load Balancing with RSRP (rLBRA): A heuristic approach whereby each UE connects to the O-RU with the highest Reference Signal Received Power (RSRP). Despite being easy and inexpensive, load imbalance often occurs due to some signal strong cells becoming congested while others remain under-loaded.

Centralized Reinforcement Learning (Centralized RL): A unified global entity situated at the Near-Real-Time RIC is responsible for making handover and load balancing decisions for all O-RUs. It relies on aggregated network state information to facilitate these decisions. This centralization allows for consideration of global conditions by the agent but limits its response to rapid local changes and increases signaling overhead.

Multi-Agent Reinforcement Learning (MARL): Every O-RU is an RL agent acting independently based on local observations without coordination. Such design is effective at local phenomena but does not result in an overall balanced loading on the network as a whole.

The choice of these baselines is based on popular strategies from the literature, which allow direct performance comparisons of different decision-making paradigms, including static heuristics, coordinated, and uncoordinated agent learning-based models.

C. Performance Metrics

Following the description of the simulation setup and the evaluated algorithms, the next step is to define the performance metrics that capture multiple aspects of network performance.

The first metric, Max-to-Min Load Difference, is used to quantify the imbalance in UE distribution across O-RUs, Let:

$\mathcal{B} = \{b_1, b_2, \dots, b_N\}$: the set of O-RUs (base stations), $\mathcal{U}(t) = \{u_1(t), u_2(t), \dots, u_M(t)\}$: the set of active UEs (users) at time t, $A_j(t) \in \mathcal{B}$: the serving O-RU for the user u_j at time t, $L_i(t)$: the load (number of connected users) on O-RU b_i at time t.

We define load per O-RU as:

$$L_i(t) = \sum_{j=1}^{M} 1_{\{A_j(t)=b_i\}} \quad (6)$$

Where $1_{\{A_j(t)=b_i\}}$ is the indicator function:

Equals 1 if user j is connected to $\mathrm{O-RU}b_i$,
Equals 0 otherwise.
Form the Load Vector. Construct the load vector at time t :

$$\mathbf{L}(t) = [L_1(t), L_2(t), \dots, L_N(t)] \tag{7}$$

This vector represents the instantaneous load distribution across all O-RUs. Then, the maximum and the minimum load for t time will be:

$$L_{\max}(t) = \max_{i\in\{1,\dots,N\}} L_i(t) \tag{8}$$

$$L_{\min}(t) = \min_{i\in\{1,\dots,N\}} L_i(t) \tag{9}$$

The instantaneous Max-to-Min Load Difference is given by:

$$\Delta_L(t) = L_{\max}(t) - L_{\min}(t) \tag{10}$$

This value captures the degree of imbalance in the user distribution at a specific moment. To evaluate performance over the entire simulation period, the average load difference is calculated as

$$\bar{\Delta}_L = \frac{1}{T}\sum_{t=1}^{T} \Delta_L(t) \tag{11}$$

While the Max-to-Min Load Difference focuses on extreme values, the Load Standard Deviation captures the overall variability in UE distribution across O-RUs. Let $L_i(t)$ denote the number of UEs connected to O-RU i at time step t, and let $\bar{L}(t)$ be the mean load across all N O-RUs at that time step, computed as

$$\bar{L}(t) = \frac{1}{N}\sum_{i=1}^{N} L_i(t) \tag{12}$$

The load variance at time t is then obtained by

$$\sigma_L^2(t) = \frac{1}{N}\sum_{i=1}^{N} (L_i(t) - \bar{L}(t))^2 \tag{13}$$

and the load standard deviation is given by

$$\sigma_L(t) = \sqrt{\sigma_L^2(t)} \tag{14}$$

To capture the overall behavior over the simulation duration, the average standard deviation is calculated as

$$\overline{\sigma_L} = \frac{1}{T}\sum_{t=1}^{T} \sigma_L(t) \tag{15}$$

In which T is the Total Number of time step. The lower values of $\overline{\sigma_L}$ mean that the loads are better balanced among the O-RUs as opposed to higher values indicating greater imbalance. This metric is a statistical view of variability and therefore provides additional information to the Max-to-Min Load Difference that uses extremes of a distribution.

Using the first two load distribution measures, Jain proposes a Fairness Index (JFI) which is a normalized measure that reflects fairness with which the total available network resources are shared by all O-RUs. Contrary to Max-to-Min Load Difference and Load Standard Deviation that use absolute values, the value produced by JFI is bounded between 0 and 1. The values closer to 1 put depicts nearly perfect fairness. Let L_i (t) represent the load of O-RU i at time step t, and N be the total number of O-RUs. At time t, Jain's Fairness Index of Jain is defined as.

$$\mathrm{JFI}(t) = \frac{\left(\sum_{i=1}^{N} L_i(t)\right)^2}{N\cdot\sum_{i=1}^{N} L_i^2(t)} \tag{16}$$

The average fairness across the simulation is then:

$$\overline{\mathrm{JFI}} = \frac{1}{T}\sum_{t=1}^{T} \mathrm{JFI}(t) \tag{17}$$

where T denotes the total number of time steps. Higher $\overline{\mathrm{JFI}}$ values reflect a more equitable distribution of UEs among O-RUs, while lower values indicate uneven resource allocation.

Overall network efficiency is evaluated using throughput, which reflects the aggregate data rate achievable under the prevailing load and channel conditions [36]. Let $r_u(t)$ denote the instantaneous throughput of user u at time step t. The network throughput at time t is:

$$R_{\mathrm{net}}(t) = \sum_{u=1}^{U} r_u(t) \tag{18}$$

where U is the number of active users. Under equal resource sharing, per-user throughput is modeled as:

$$r_u(t) = \eta B \log_2(1 + \mathrm{SINR}_u(t)) \cdot \frac{1}{n_{i(u,t)}(t)} \tag{19}$$

where η is the efficiency factor, B is the available bandwidth, and $n_{i(u,t)}(t)$ is the number of users connected to the serving $\mathrm{O-RU}i(u,t)$.

The average network throughput over T time steps is:

$$\bar{R}_{\mathrm{net}} = \frac{1}{T}\sum_{t=1}^{T} R_{\mathrm{net}}(t) \tag{20}$$

Higher $\bar{R}_{\mathrm{net}}$ indicates better spectral efficiency under realistic sharing; Usually, the improvements occur when the SINR is enhanced. In addition, bottleneck or congestion is avoided via load balancing.

Another metric to calculate is the success rate (HSR) of handovers, which is the percentage of such attempts completed successfully without service interruption. We define N_{succ} as the number of successful handovers during the simulation and N_{total} as the total number of handovers attempted. Method of Calculating HSR.

$$\mathrm{HSR} = \frac{N_{\mathrm{succ}}}{N_{\mathrm{total}}} \tag{21}$$

The range of this metric is 0 to 1. Higher values are more reliable for maintaining a connection while mobile. A high HSR indicates that the handoff decision process, signaling, and resource allocation all function effectively in practice.

The repeated hand-offs of users between cells may lead to inefficient mobility, even when the cell is quite successful. This term refers to the Ping-Pong Rate (PPR). To allow for its calculation, let N_{pp} denote the total number of detected ping-pong events and N_{succ} the total number of successful handovers. The measurement has the following:

$$\mathrm{PPR} = \frac{N_{\mathrm{pp}}}{N_{\mathrm{succ}}} \tag{22}$$

PPR low values denote more stable handover decisions as users often do not oscillate between cells unnecessarily. In

balanced networks, the ping-pong rate is likely to be low when the success rate is high for proper mobility management.

V. RESULTS AND DISCUSSION

The proposed FedRL-O-RAN framework evaluation is structured on 2 categories of performance indicators. The first classification aims to assess the efficiency of load balancing using parameters such as Max-to-Min Load Difference, Load Standard Deviation, and Jain's Fairness Index (JFI), which measure the uniformity and temporal stability of user distributions across O-RUs. The 2nd category is user-experience quality. This is measured using Throughput, Handover Success Rate (HRS), and Ping-Pong Rate (PPR). The first metric we consider is the maximum-to-minimum load difference, which measures the difference in UE load distribution across O-RUs at each time step. A low value indicates a more balanced load across the network, whereas a high value indicates severe congestion in some cells and underutilization in others. In an O-RAN environment, when load distribution is uneven, spectral efficiency decreases. Moreover, Quality of Service (QoS) also suffers due to uneven loads. Figure 1 shows the allocation of the distributed resource for single- and mixed-load types.

The figure shows a small average difference and a tight variability range, indicating a consistently balanced distribution of the UE. On the other hand, the static, signal-strength based association policy of rLBRA produces the widest spread. Although centralized RL and MARL yield moderate gains, they still occasionally experience imbalances. The FedRL remains stable because it builds on locally adaptable decisions, followed by a global model sync to handle congestion.

FIGURE 1 the boxplot for the Max-to-Min Load Difference across all evaluated approaches

We also analyze the Load Standard Deviation, which reflects the overall variability in load allocation across the O-RUs over time in addition to extreme-value differences. Lower values indicate enduring fairness and minor fluctuations, which are vital for sustaining network performance. Figure 2 displays comparative boxplots illustrating the Load standard deviation. Our framework has the lowest average standard deviation, coupled with a very compact interquartile range, indicating a stable, properly distributed load. The rLBRA methodology exhibits the greatest variability due to its frequent overload of strong signal cells.

FIGURE 2 the comparative boxplot for the Load Standard Deviation among the evaluated methods

Models for single-agent forecasting of future service supply and demand have been developed. The special characteristic of FedRL is that it can respond quickly to local model change while synchronously and coherently updating the global policy. It preserves transient stability when a delay in the onset of load increase occurs. To measure the fairness in the allocation of resources Jain's Fairness Index (JFI), which is a normalized measure and varies from 0 to 1, with closer to 1 being more fair. With high fairness, no particular subset of users has consistently worse performance (Figure 3).

The fairness values of the proposed FedRL algorithm exceed 0.95 for all simulation sets. Thus, FedRL can produce more reliable fairness values than the baseline methods. The rLBRA method results in the lowest fairness values. This is because the static, signal-strength-driven association overutilizes some O-RUs and underutilizes others. While centralized reinforcement learning (RL) enhances rLBRA, its response to localized variations is sluggish.

In contrast, multi-agent RL (MARL) allows local adaptation but lacks coordination between agents, leading to unequal outcomes. FedRL is superior and fairer than prior work due to its two-tier architecture. In particular, it achieves local learning sensitive to the cell's conditions, while the global model aligns resource-sharing strategies across the network.

FIGURE 3 JFI distribution for all compared methods

While these performance results reinforce FedRL's ability to achieve even, stable load distribution, it is noteworthy that load balancing alone cannot guarantee a good end-user experience. Consequently, the evaluation of user-experience metrics, Throughput, Handover Success Rate (HSR) and Ping-Pong Rate (PPR) will be done to find out the practical advantage of balanced resources.

Throughput is defined as the total data rate of all active users under prevailing load and channel conditions. Higher throughput means better spectrum utilization and better quality of service (QoS), while stability over time indicates robustness against mobility and traffic.

As shown in figure 4, the practical impact of load balancing on a given metric can be easily gauged. Throughput assessment results indicate that the median values for FedRL were the highest while exhibiting the lowest variability, showcasing its broad adaptability. The rLBRA method has the least performance and shows a lower median and higher variance efficiency due to congestion in strong signal cells. Centralized RL and MARL achieve moderate performance improvements, with MARL slightly better due to faster adaptation to local changes. FedRL's better performance is due to its balanced load distribution, reduced unnecessary handovers, and consistent maintenance of good SINR values.

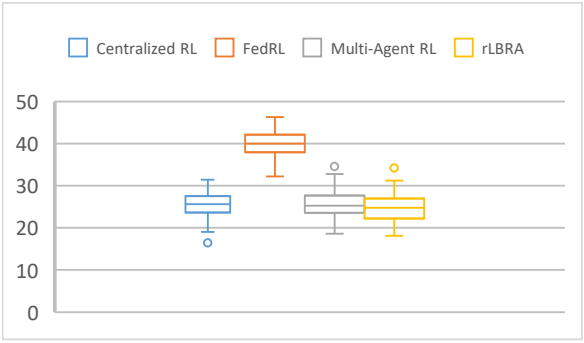


FIGURE 4 comparative throughput distributions for the evaluated methods.

Handover Success Rate is the ratio of handover attempts completed successfully without service interruption. The higher the value, the more reliable the mobility management; the lower it is, the more likely it may signal problems with decision-making, signalling, or resource assignment during handovers. In dynamic O-RAN deployments, consistently high HSR is essential to maintain user connectivity without interruption. According to Figure 5, this metric allows direct comparison of the reliability of different decision-making strategies.

According to HSR results, FedRL maintained values of 85% to 88%, the highest among the compared schemes, while rLBRA, Centralized RL, and MARL cluster at 75% to 80%. Thus, it can be inferred that rLBRA's limited performance stems from its static association policy, which fails to account for real-time mobility and load variations. Although MARL and Centralized RL improved on reactive agents, neither used advanced hyperparameter tuning (Centralized RL) or coordination among agents (MARL). FedRL combines a context-aware local decision-making process with an infrequent global policy update mechanism. Thus, timely and stable handover execution becomes possible with lower failure rates.

The Ping-Pong Rate (PPR) quantifies the proportion of handovers that quickly reverse, returning a UE to its previous serving cell within a short interval. High PPR values indicate instability in mobility management, leading to unnecessary signaling, increased latency, and degraded quality of experience. Reducing this metric is essential to ensuring smooth, efficient handover operations. As illustrated in Figure 6, PPR provides a measure of handover stability across the evaluated methods.

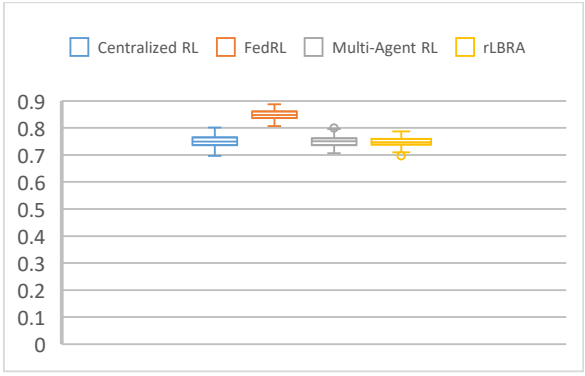


FIGURE 5 comparative HSR performance for all evaluated methods

FIGURE 6 The comparative PPR results for all evaluated methods

According to the PPR analysis, we achieve the lowest median value with the most compact distribution. This indicates that our handover decisions are stable and well-judged. The rLBRA method shows the most ping-pong effect, caused by a static signal-threshold trigger that reacts to transient fluctuations and does not take the user trajectory into account. MARL outperforms Centralized RL on this parameter, as it effectively adapts to local mobility patterns. However, it lacks coordination, making it less consistent than FedRL. The edge of the FedRL over other methods is its own reward function, which discourages instability in Handover. Other is the O-RU of neighboring it, aligned in policy, so it helps avoid discordant action. The recommended FedRL-O-RAN framework naturally

aligns with the Near-RT RIC architecture in the O-RAN standards from a deployment viewpoint. The signaling overhead for federated updates is still relatively modest, as each O-RU only needs to send its compact Q-table rather than raw user data, as in centralized state collection. In addition, the aggregation function at the Near-RT RIC consists of simple averaging operations, which scale linearly with the number of O-RUs that join the process. As a result, it leads to a low computation load that can be adopted in real-time RIC. As more devices are connected to the network, hierarchical or clustered aggregation approaches can further enhance scalability.

To evaluate the scalability of the proposed framework, the deployment scenarios summarized in Table 2 were evaluated. By varying the number of O-RUs and the deployment area, the analysis examines whether FedRL can maintain its load-balancing and mobility-management performance as the network size increases.

The scalability results presented in Table 3 show that the proposed framework remains effective across all evaluated scenarios. While larger deployment areas increase mobility complexity and user traversal distances, a higher number of O-RUs creates a denser handover environment. Despite these challenges, the federated learning mechanism maintains policy consistency across distributed agents, resulting in stable fairness, handover reliability, and load-balancing performance.

Table 3 summarizes the scalability performance of the proposed FedRL framework under different deployment areas and O-RU densities. The results show that FedRL maintains a high level of fairness across all evaluated scenarios, with Jain's Fairness Index remaining above 0.95. Likewise, the handover success rate remains close to 88%, indicating that the learned policy continues to make effective mobility-management decisions as the network scales.

TABLE 3
SCALABILITY PERFORMANCE SUMMARY

Scenario	O-RUs	JFI ↑	HSR ↑	PPR ↓	Load Difference ↓
S1	7	0.968	0.879	0.032	8.04
S2	14	0.964	0.881	0.031	8.71
S3	7	0.962	0.876	0.034	9.12
S4	14	0.959	0.878	0.029	9.93

The load difference shows only a modest increase as the deployment area and O-RU density increase, indicating that the proposed federated learning framework continues to balance traffic effectively across neighboring O-RUs. Similarly, the ping-pong rate remains consistently low across all scenarios. Overall, the results demonstrate that the proposed framework scales effectively while preserving the fairness, mobility-management, and load-balancing benefits achieved in the baseline deployment.

VI. CONCLUSIONS

The paper presents FedRL-O-RAN, a decentralized load-balancing and handover-optimization framework for O-RAN-enabled 5G and beyond networks. The use of distributed reinforcement learning via Q-learning agents at every O-RU which enables the agents to make context-aware decisions locally. Furthermore, periodic federated averaging enables the agents to coordinate their decision-making. Thus, this scheme facilitates local context-aware decisions while maintaining global policy alignment and data privacy. In a modelled multi-cell O-RAN topology, FedRL achieved a reduction of more than 50% in load imbalance compared to rule-based schemes and 25% compared to centralized RL. Jain's Index consistently indicates fairness above 0.95. The framework's use further improved throughput by more than 15%, increased handover success rates by nearly 13% compared to the rule-based method, and reduced ping-pong events by 30 to 40%, thereby improving stability and signaling overhead. These benefits indicate that the system can provide balanced load distribution, fairness, and quality of service despite varying mobility and traffic. The proposed solution incorporates proven architectures and principles to build a scalable, agile foundation for intelligent RAN automation, while leveraging the flexibility and openness of next-generation architectures. In addition to showing improved performance, this project shows a practical path to integrate federated reinforcement learning into control loops using Near-RT RIC. The suggested architecture can be used for more detailed coordination tasks, such as multi-objective optimization, energy-aware control, and slice-level orchestration, in B5G and 6G systems. Additional scalability experiments showed that the proposed FedRL framework continues to deliver effective load balancing and reliable mobility management as the deployment area and O-RU density increase. In the future, research can explore hierarchical federated aggregation (HFA), non-IID traffic scenarios, and integration with advanced antenna and beamforming models.

REFERENCES

[1] N. Q. A. AlShaikhli, M. A. Neamah, and M. I. Aal-Nouman, "Design of triband antenna for telecommunications and network applications," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 2, pp. 1084–1090, 2022. DOI: 10.11591/eei.v11i2.3233

[2] C. Adamczyk and A. Kliks, "Conflict mitigation framework and conflict detection in O-RAN near-RT RIC," *IEEE Communications Magazine*, vol. 61, no. 12, pp. 199–205, 2023. DOI: 10.1109/MCOM.018.2200752

[3] O-RAN Alliance, "O-RAN: Towards an Open and Smart RAN," White paper, Aug. 2018.

[4] M. Polese, L. Bonati, S. D'oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, 2023. DOI: 10.1109/COMST.2023.3239220

[5] W. Azariah, F. A. Bimo, C.-W. Lin, R.-G. Cheng, N. Nikaein, and R. Jana, "A Survey on Open Radio Access Networks: Challenges, Research Directions, and Open Source Approaches," *Sensors*, vol. 24, no. 3, 2024, DOI: 10.3390/s24031038.

[6] K. Alam et al., "A Comprehensive Tutorial and Survey of O-RAN: Exploring Slicing-Aware Architecture, Deployment Options, Use Cases, and Challenges," in *IEEE Communications Surveys & Tutorials*, vol. 28, pp. 1637–1678, 2025, DOI: 10.1109/COMST.2025.3598406

[7] Q. Sun, N. Li, C. I. -L., J. Huang, X. Xu, and Y. Xie, "Intelligent RAN Automation for 5G and Beyond," *IEEE Wirel Commun*, vol. 31, no. 1, pp. 94–102, 2024, DOI: 10.1109/MWC.014.2200271.

[8] R. Ntassah, G. M. Dell'Aera, and F. Granelli, "PPO-EPO: Energy and Performance Optimization for O-RAN Using Reinforcement Learning," arXiv preprint. DOI: 10.48550/arXiv.2504.14749

[9] A. Almeida, P. Rito, S. Brás, F. C. Pinto, and S. Sargento, "A machine learning approach to forecast 5G metrics in a commercial and operational 5G platform: 5G and mobility," *Comput Commun*, vol. 228, p. 107 974, 2024. DOI: 10.1016/j.comcom.2024.107974

[10] E. Gures, I. Shayea, M. Ergen, M. H. Azmi, and A. A. El-Saleh, "Machine learning-based load balancing algorithms in future heterogeneous networks: A survey," I*EEE Access*, vol. 10, pp. 37 689– 37717, 2022. **DOI**: 10.1109/ACCESS.2022.3161511

[11] J. Ochoa-Aldeán, C. Silva-Cárdenas, R. Torres, J. I. Gonzalez, and S. Fortes, "Algorithms for Load Balancing in Next-Generation Mobile Networks: A Systematic Literature Review," *Future Internet*, vol. 17, no. 7, p. 290, 2025. **DOI**: 10.3390/fi17070290

[12] R. K. Mohammed and N. N. Khamiss, "Hybrid Multiple Access Techniques Performance Analysis Of Dynamic Resource Allocation," *Iraqi Journal of Information and Communication Technology*, vol. 7, no. 1, pp. 23–34, 2024. **DOI**: 10.31987/ijict.7.1.243

[13] Kiran, Kotaru, and D. Rajeswara Rao. "Analytical review and study on various vertical handover management technologies in 5G heterogeneous network." *Infocommunications Journal 14*, no. 2 (2022): 28-38. **DOI**: 10.36244/ICJ.2022.2.3

[14] N. Saqib, N. F. Abdullah, A. Abu-Samah, H. A. H. Alobaidy, and R. Nordin, "Adaptive VNF Placement Considering Overall Latency and 5G Wireless Channel Reliability in Industry 4.0: A Reinforcement Learning Based Approach," *IEEE Access*, vol. 12, pp. 88883–88896, 2024, **DOI**: 10.1109/ACCESS.2024.3419065.

[15] O. Orhan, V. N. Swamy, T. Tetzlaff, M. Nassar, H. Nikopour, and S. Talwar, "Connection management xAPP for O-RAN RIC: A graph neural network and reinforcement learning approach," in *2021 20th IEEE international conference on machine learning and applications (ICMLA)*, IEEE, 2021, pp. 936–941. **DOI**: 10.1109/ICMLA52953.2021.00154.

[16] C.-H. Lai, L.-H. Shen, and K.-T. Feng, "Intelligent load balancing and resource allocation in O-RAN: A multi-agent multi-armed bandit approach," in *2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, 2023, pp. 1–6. **DOI**: 10.1109/PIMRC56721.2023.10293795

[17] H. Zhang, H. Zhou, and M. Erol-Kantarci, "Federated deep reinforcement learning for resource allocation in O-RAN slicing," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*, IEEE, 2022, pp. 958–963. **DOI**: 10.1109/GLOBECOM48099.2022.10001658

[18] B. Zhao, Q. Cui, W. Ni, X. Li, and S. Liang, "Multi-Layer Collaborative Federated Learning architecture for 6G Open RAN," *Wireless Networks*, vol. 31, no. 2, pp. 1377–1390, 2025. **DOI**: 10.1007/s11276-024-03823-0

[19] M. Behjati, H. A. H. Alobaidy, R. Nordin, and N. F. Abdullah, "UAV-assisted federated learning with hybrid LoRa P2P/LoRaWAN for sustainable biosphere," *Frontiers in Communications and Networks*, vol. Volume 6-2025, 2025, **DOI**: 10.3389/frcmn.2025.1529453.

[20] M. Q. Hamdan et al., "Recent Advances in Machine Learning for Network Automation in the O-RAN," *Sensors*, vol. 23, no. 21, 2023, **DOI**: 10.3390/s23218792.

[21] Y. Azimi, S. Yousefi, H. Kalbkhani, and T. Kunz, "Mobility aware and energy-efficient federated deep reinforcement learning assisted resource allocation for 5G-RAN slicing," *Comput Commun*, vol. 217, pp. 166– 182, 2024, **DOI**: 10.1016/j.comcom.2024.01.028.

[22] S. Gu, C. You, B. Ren, and D. Guo, "Communication and Computation Efficient Split Federated Learning in O-RAN," arXiv preprint 2025. **DOI**: 10.48550/arXiv.2508.02534

[23] T. Pamuklu, M. Erol-Kantarci, and C. Ersoy, "Reinforcement learning based dynamic function splitting in disaggregated green open RANs," in *ICC 2021-IEEE International Conference on Communications*, IEEE, 2021, pp. 1–6. **DOI**: 10.1109/ICC42927.2021.9500721

[24] Q. Ali, "Performance Optimization and Architectural Advancements in Cloud Radio Access Networks (C-RAN) for 5G and Beyond," *Journal of Electronic & Information Systems| Volume*, vol. 7, no. 01, 2025. **DOI**: 10.30564/jeis.v7i1.9960

[25] T. S. Kumar et al., "Advanced Handover Optimization (AHO) using deep reinforcement learning in 5G Networks," *Journal of King Saud University Computer and Information Sciences*, vol. 37, no. 6, pp. 1–14, 2025. **DOI**: 10.1007/s44443-025-00124-0

[26] M. Gupta, R. M. Dreifuerst, A. Yazdan, P.-H. Huang, S. Kasturia, and J. G. Andrews, "Load balancing and handover optimization in multi-band networks using deep reinforcement learning," in *2021 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2021, pp. 1–6. **DOI**: 10.1109/GLOBECOM46510.2021.9685781

[27] P. Sroka, Ł. Kułacz, S. Janji, M. Dryjański, and A. Kliks, "Policy-based traffic steering and load balancing in o-ran-based vehicle-to-network communications," *IEEE Trans Veh Technol*, vol. 73, no. 7, pp. 9356–9369, 2024.

[28] H. Lin, Y.-K. Su, H.-Q. Chen, and L.-F. Ko, "Federated Deep Q-Learning and 5G load balancing," *arXiv preprint arXiv:2403.08813*, 2024. **DOI**: 10.48550/arXiv.2403.08813

[29] Y. Cao, S.-Y. Lien, Y.-C. Liang, K.-C. Chen, and X. Shen, "User access control in open radio access networks: A federated deep reinforcement learning approach," *IEEE Trans Wirel Commun*, vol. 21, no. 6, pp. 3721–3736, 2021. **DOI**: 10.1109/TWC.2021.3123500

[30] F. Ahmed et al., "Federated Deep Reinforcement Learning-Driven O-RAN for Automatic Multirobot Reconfiguration" *NOMS 2025 IEEE Network Operations and Management Symposium*, Honolulu, HI, USA, 2025, pp. 1–7, **DOI**: 10.1109/NOMS57970.2025.11073682.

[31] P. Abdisarabshali, N. Accurso, F. Malandra, W. Su, and S. Hosseinalipour, "Synergies between federated learning and o-ran: Towards an elastic virtualized architecture for multiple distributed machine learning services," arXiv preprint 2023. **DOI**: 10.48550/arXiv.2305.02109.

[32] V.-D. Nguyen et al., "Network-aided intelligent traffic steering in 6G O-RAN: A multi-layer optimization framework," *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 2, pp. 389–405, 2023. **DOI**: 10.1109/JSAC.2023.3336183.

[33] F. Kavehmadavani, V.-D. Nguyen, T. X. Vu, and S. Chatzinotas, "Intelligent traffic steering in beyond 5G open RAN based on LSTM traffic prediction," *IEEE Trans Wirel Commun*, vol. 22, no. 11, pp. 7727–7742, 2023. **DOI**: 10.1109/TWC.2023.3254903

[34] C. Sun, Y. Zhou, G. Jung, T. X. Tran, and D. Pompili, "Carl: Cascade reinforcement learning with state space splitting for o-ran based traffic steering," *arXiv preprint arXiv:2312.01970*, 2023.

[35] O. Semiari, H. Nikopour, and S. Talwar, "*Graph Reinforcement Learning for QoS-Aware Load Balancing in Open Radio Access Networks*," arXiv e-prints, p. arXiv-2504, 2025. **DOI**: 10.1109/ICCWorkshops67674.2025.11162460

[36] Raya Adel Kamil , Saif Mohamed Baraa Alsabti, Rusul K. Abdulsattar, Ammar H. Mohammed, and Taha A. Elwi "On the Enhancement Anomaly Detection for RF Bio-Sensors by Computing Artificial Networks Using Machine Learning Techniques", *Infocommunications Journal*, Vol. XVII, No 2, June 2025, pp. 89-95., **DOI**: 10.36244/ICJ.2025.2.11

Mohammed I. Aal-Nouman (Member, IEEE) received his Ph.D. in Telecommunications Engineering from the University of Salford, Manchester, UK, in 2017. He got his MSc. In Computer Engineering from Al-Nahrain University in 2007. He is currently a Lecturer in the College of Information Engineering at Al-Nahrain University, Baghdad, where he has served since 2008. His research interests include mobile and wireless communications, heterogeneous networks, and performance optimization in next-generation communication systems.

Sarmad M. Hadi is a lecturer at the College of Information Engineering, Al-Nahrain University, Iraq, and the head of the university's computer center. He received his M.Sc. degree from Al-Nahrain University, Iraq (2010), and his Ph.D. degree in Information and Communication Engineering from Al-Nahrain University (2020). His research interests include AI, web applications, multimedia, bioinformatics and modern communication technologies.

Haider A. H. Alobaidy (Member, IEEE) is a lecturer at the College of Information Engineering, Al-Nahrain University, Iraq. Prior to his current role, he served as a postdoctoral researcher at Universiti Kebangsaan Malaysia (UKM), where he further honed his expertise in wireless communication systems and related fields. He received an M.Sc. degree from Al-Mustansiriyah University, Iraq (2016), and a Ph.D. degree in Electrical, Electronics, and Systems Engineering from UKM (2022). His research interests include wireless communication, wireless sensor networks, IoT, machine learning, and channel propagation modeling and estimation.

Convolutional and Variational Autoencoders with Supervised Classifiers for RF Spectrum Anomaly Detection

Szilárd László Takács*, and Konrád Kajdy

***Abstract*—Effective and secure monitoring of the radio frequency spectrum is essential for the reliable operation of modern wireless communication systems. Automated detection of spectral anomalies can support regulatory and operational monitoring tasks. This study evaluates three autoencoder architectures — Vanilla Autoencoder (AE), Convolutional Autoencoder (CAE), and Variational Autoencoder (VAE) — for anomaly detection using waterfall image representations derived from FM-band (86.5–108 MHz) measurement data. The models were assessed both as standalone reconstruction-based detectors and as feature extractors combined with supervised classifiers. Standalone autoencoders achieved F1-scores between 0.794 and 0.813. Higher performance was obtained when encoder-derived latent representations were used with supervised models, where the best result (F1-score: 0.915) was achieved by the CAE + ExtraTrees combination. The results indicate that hybrid encoder–classifier approaches can provide an effective practical solution for anomaly detection in spectrum monitoring environments. While promising, the findings are limited to the investigated FM-band dataset and require further validation across broader spectrum environments.**

***Index Terms*—spectrum monitoring; autoencoder; anomaly detection; RF spectrum anomalies**

I. INTRODUCTION

Ensuring uninterrupted wireless communication has become increasingly challenging due to the growing number of radio systems and devices sharing the spectrum. Interference, unauthorized emissions, and technical malfunctions may affect multiple frequency bands, including the frequency modulated (FM) radio band (86.5–108 MHz). Consequently, efficient monitoring tools are required to support timely anomaly detection and response.

In Hungary, wireless communication is regulated under Act C of 2003 on Electronic Communications [1]. Spectrum supervision is supported by a nationwide monitoring infrastructure consisting of 67 measurement stations. In current operational practice, substantial parts of anomaly assessment still require expert interpretation. As spectrum usage becomes increasingly dynamic, automated decision-support methods may improve efficiency and consistency.

Szilárd László Takács Department of Electrical Engineering and Infocommunications, Széchenyi István University, Győr, Hungary; (e-mail: takacs.szilard.laszlo@sze.hu).

Konrád Kajdy Széchenyi István University, Győr, Hungary; (e-mail: kajdykonrad@gmail.com).

Recent advances in machine learning have created new opportunities for supporting automated RF spectrum analysis. In particular, image-based representations such as waterfall diagrams enable the application of computer vision and representation-learning methods to spectrum monitoring tasks. However, comparative evidence on how different autoencoder architectures perform on real FM-band monitoring data remains limited.

The contribution of this study is primarily empirical. First, three commonly used autoencoder architectures are compared on real FM-band waterfall measurements. Second, the usefulness of combining encoder-derived latent representations with conventional supervised classifiers is examined. The goal is not to introduce a novel architecture, but to evaluate a practical hybrid pipeline for operational RF anomaly detection.

II. RELATED WORK

Anomaly detection in RF spectrum monitoring is inherently challenging because abnormal events may vary substantially in frequency, duration, intensity, and temporal structure. Frequently recurring and well-defined events, such as transmission start/stop patterns, persistent level changes, or known interference types, can often be addressed using supervised machine learning methods when labeled examples are available. However, rare, previously unseen, or weakly characterized anomalies remain more difficult to model in a purely supervised setting. For this reason, unsupervised representation-learning approaches, including autoencoders, continue to receive attention.

The application of machine learning to radio spectrum analysis is an emerging research area. Prior studies have investigated unsupervised spectrum anomaly detection, spectrum sensing, and interference recognition. The SAIFE framework, based on an adversarial autoencoder, demonstrated anomaly detection and localization using power spectral density data [2]. Stacked autoencoder models have also been applied to OFDM spectrum sensing, showing robustness against noise and synchronization uncertainties [3]. These studies indicate that latent-feature learning can be useful when explicit anomaly labels are limited. Beyond RF applications, autoencoder-based methods have also been studied in adjacent signal and image domains, including hyperspectral image classification and industrial fault detection

DOI: 10.36244/ICJ.2026.2.7

[4]–[7]. Although these domains differ from RF monitoring, they provide evidence that learned low-dimensional representations may improve downstream classification tasks.

At the same time, several limitations of reconstruction-based anomaly detection have been reported. Prior work has shown that anomalous samples may sometimes be reconstructed with low error, which can reduce detection sensitivity [8]. Dataset topology may also affect reconstruction behavior [9], while contaminated training data can degrade robustness [10]. In addition, adversarial manipulation has been identified as a potential deployment risk [11]. These findings suggest that reconstruction error alone may be insufficient in some practical scenarios.

Motivated by these limitations, hybrid approaches that combine unsupervised feature extraction with supervised classification may offer a more reliable alternative. However, comparative studies evaluating multiple autoencoder architectures on real FM-band waterfall measurements remain limited. This study addresses that gap by comparing vanilla, convolutional, and variational autoencoders both as standalone detectors and as feature extractors for downstream classifiers.

More recent studies have also explored convolutional neural networks and transformer-inspired architectures for modulation recognition, interference classification, and wideband spectrum sensing, indicating a broader shift toward end-to-end learned RF representations [12], [13].

Unlike prior studies focusing on single-model designs, this work emphasizes comparative benchmarking under identical measurement conditions

III. Materials and Methods

This section describes the measurement dataset, preprocessing workflow, model architectures, training configuration, and evaluation protocol used in the study. Particular attention was given to preventing data leakage and obtaining robust performance estimates despite the limited dataset size.

A. Data Collection and Dataset Construction

Data collection was carried out at two stations of the Hungarian national spectrum monitoring network, located in Dobogókő and Kisvárda, within the FM broadcast band (86.5–108 MHz). Waterfall diagrams were generated from the measurements, where the x-axis represents frequency, the y-axis represents time (400-second intervals), and the color scale represents received signal strength.

To ensure consistency across stations, measured values were linearly rescaled to the interval −20 to 120 dBμV. The measurement frequency resolution was 5 kHz.

The resulting dataset contained 1,787 training images and 280 evaluation images. The hold-out evaluation set contained 135 anomalous and 145 normal samples, while the training partition consisted only of samples not appearing in the evaluation subset. Because the available dataset size is moderate for deep learning applications, regularization, augmentation, and cross-validation were incorporated into the evaluation pipeline.

B. Data Preprocessing and Annotation

Anomaly labels were assigned through expert inspection by multiple spectrum monitoring specialists. Independent assessments were reconciled through consensus review to reduce subjectivity.

From the original waterfall diagrams, image patches of 640 × 640 pixels were extracted and resized to 128 × 128 pixels for model input. Data augmentation included random horizontal and vertical flips, brightness/contrast variation, and normalization.

No synthetic anomalies were introduced. No explicit denoising was applied, as preserving naturally occurring RF noise patterns was considered important for operational realism.

C. Autoencoder Architectures

Three encoder–decoder architectures were evaluated: a fully connected autoencoder (AE), a convolutional autoencoder (CAE), and a variational autoencoder (VAE). These models were selected to compare deterministic dense representations, spatially aware convolutional representations, and probabilistic latent representations within a common experimental framework.

- Vanilla AE: A symmetric, fully connected neural network that reduces the 49,152-dimensional input space to a 512-dimensional latent representation through successive layers of 4096, 2048, and 1024 neurons using ReLU activation. The decoder mirrors the encoder in reverse, with a sigmoid output activation ensuring reconstructions remain in the [0,1] range.
- CAE: A convolutional architecture comprising five sequential Conv2D layers (32–256 filters, 4×4 kernels, stride = 2, padding = 1) with ReLU activations. Compression is finalized by an Adaptive Average Pooling layer, followed by a linear projection into a 128-dimensional latent space. Reconstruction is achieved through transposed convolutional layers, gradually reducing the filter size from 256 to 32, and finalized with a sigmoid activation.
- VAE: Architecturally similar to the CAE encoder, but the latent space is modeled as a probabilistic distribution. Two fully connected layers output the mean and log-variance of the latent space, from which a 128-dimensional latent vector is sampled. The decoder is identical to that of the CAE.

D. Model Training Procedure

All autoencoder models were trained for 50 epochs using the Adam optimizer with a learning rate of 0.001 and a batch size of 64. Mean squared error (MSE) was used as the reconstruction loss for all autoencoder architectures. Model parameters were updated through backpropagation. Training was performed on GPU when CUDA was available.

After training, only the encoder components were retained and used to generate latent feature representations. These extracted features were then used as inputs for supervised classifiers (Random Forest, Support Vector Machine, ExtraTrees, Multi-Layer Perceptron, and others), each optimized using its standard learning objective.

To reduce overfitting risk, data augmentation and held-out evaluation were applied. Future work may investigate larger datasets and early stopping strategies.

E. Feature Extraction

After training, the encoder components were separated and used as feature extractors. This produced latent representations of dimension 512 for AE and 128 for both CAE and VAE.
These compact representations were then used as input features for conventional supervised classifiers.

F. Supervised Classifiers

The following supervised models were evaluated: Random Forest, Gradient Boosting, XGBoost, Logistic Regression, Support Vector Machine, k-Nearest Neighbours, Gaussian Naive Bayes, ExtraTrees, Multi-Layer Perceptron, and LightGBM.
This diverse set was selected to compare linear, kernel-based, ensemble, probabilistic, and neural classification paradigms.

G. Validation Strategy

The hold-out test set was fixed prior to model development and remained untouched until final evaluation.
Autoencoder models were trained on a fully separate unsupervised training subset. Supervised classifiers were then trained on encoder-derived latent representations using a distinct labeled training subset. No samples, overlapping waterfall segments, or duplicated measurements were shared across the autoencoder training, classifier training, and hold-out test partitions.
Five-fold cross-validation was applied only within the supervised classifier training subset for model selection and hyperparameter tuning. Final performance metrics were computed once on the independent hold-out test set.

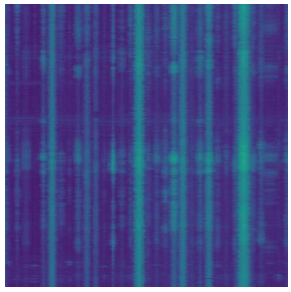


Fig. 1. Example of a waterfall diagram in the FM frequency band (86.5 – 108 MHz). The x-axis represents frequency, the y-axis represents time, and the color scale encodes signal strength.

TABLE I
THE BEST AND AVERAGE PERFORMANCE OF SUPERVISED CLASSIFIERS ON THE TEST SET FOR EACH AUTOENCODER ARCHITECTURE.

Autoencoder	AE	CAE	VAE
Classifier	SVM	ExtraTrees	MLP
Accuracy	0.893	0.911	0.893
Precision	0.839	0.844	0.862
Recall	0.963	1	0.926
F1-score	0.897	0.915	0.893
ROC AUC	0.861	0.966	0.950

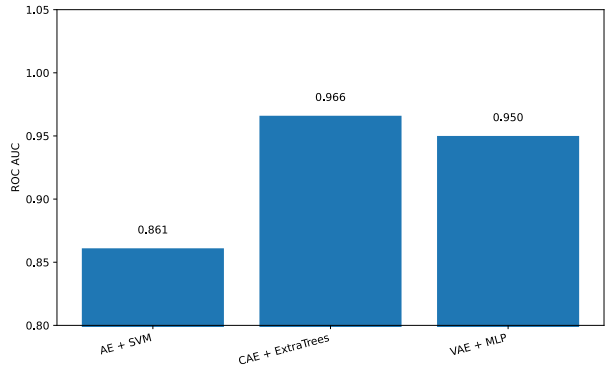


Fig. 2. ROC AUC of the best classifier–encoder combinations: AE + SVM, CAE +ExtraTrees, VAE + MLP.

IV. RESULTS

This section summarizes the empirical performance of standalone autoencoders and hybrid encoder–classifier pipelines on the evaluated FM-band anomaly detection task.

A. Autoencoder-Based Reconstruction Performance

The three autoencoder variants were first evaluated as reconstruction-based anomaly detectors. All models achieved comparable performance, with F1-scores ranging from 0.794 to 0.813.
Among the standalone approaches, the VAE produced the strongest result, followed closely by the vanilla AE, while the CAE achieved slightly lower performance. These findings suggest that latent-space regularization may provide modest benefits, although standalone reconstruction-based detection remained limited overall.

B. Feature Extraction and Supervised Classification

Substantial improvements were observed when encoder outputs were used as feature representations for supervised learning. Across all three encoder families, multiple classifiers outperformed the corresponding standalone autoencoders.
The highest measured performance within the evaluated configurations was obtained by the CAE + ExtraTrees combination, which achieved the highest F1-score and recall among the evaluated models. The AE + SVM pipeline also performed strongly, particularly in sensitivity-oriented detection scenarios, while VAE + MLP yielded the most balanced trade-off across evaluation metrics.
Relative to standalone autoencoders, the hybrid pipelines improved F1-score by approximately 10.3% (AE), 15.2% (CAE), and 10.0% (VAE).
Detailed numerical results are summarized in Tables I–III, while ROC-based comparisons are presented in Fig. 2 and aggregate metric comparisons in Figs. 3–5.

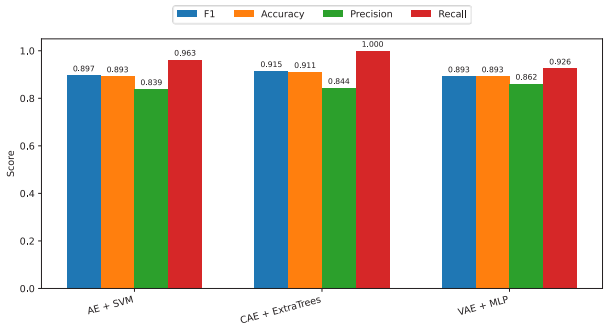


Fig. 3. Comparison of F1, Accuracy, Precision, and Recall for the best combinations.

TABLE II
MEAN F1-SCORE OF SUPERVISED CLASSIFIERS ON THE VALIDATION SET FOR EACH AUTOENCODER ARCHITECTURE.

Classifier	AE	CAE	VAE
RandomForest	0.840 ± 0.015	0.798 ± 0.040	0.848 ± 0.025
GradientBoosting	0.858 ± 0.025	0.818 ± 0.018	0.864 ± 0.039
XGBoost	0.828 ± 0.041	0.825 ± 0.042	0.864 ± 0.039
LogisticRegression	0.818 ± 0.039	0.777 ± 0.068	0.858 ± 0.026
SVM	0.856 ± 0.023	0.781 ± 0.045	0.875 ± 0.031
kNN	0.852 ± 0.021	0.813 ± 0.033	0.847 ± 0.039
GaussianNB	0.803 ± 0.051	0.805 ± 0.051	0.816 ± 0.045
ExtraTrees	0.831 ± 0.025	0.817 ± 0.045	0.863 ± 0.053
MLP	0.841 ± 0.063	0.807 ± 0.068	0.874 ± 0.042
LightGBM	0.805 ± 0.030	0.823 ± 0.025	0.856 ± 0.027

V. DISCUSSION

The results indicate that combining encoder-derived latent representations with supervised classifiers appears to be an effective strategy for RF spectrum anomaly detection on the evaluated FM-band dataset. In all three investigated encoder families, hybrid pipelines outperformed the corresponding standalone reconstruction-based autoencoders, suggesting that latent features contained discriminative information not fully exploited by reconstruction error alone.

Among the evaluated configurations, the CAE + ExtraTrees model achieved the best performance among the evaluated configurations. This may indicate that convolutional encoders are well suited to capturing local spatial structures present in waterfall images, while ensemble classifiers can effectively separate normal and anomalous feature patterns. The VAE + MLP configuration produced a competitive and balanced alternative, while AE + SVM also yielded strong sensitivity-oriented performance.

These findings are consistent with prior studies showing that hybrid representation-learning pipelines can be useful when labeled anomaly data are limited [2]–[6]. Direct CNN baselines were outside the scope of this initial benchmarking study and are reserved for future work. In practical monitoring environments, such approaches may offer operational advantages because feature extraction and classification can be updated independently.

At the same time, the results should be interpreted within the scope of the present study. The experiments were conducted on a relatively limited dataset size for deep learning applications, restricted to the FM broadcast band, and based on measurements from two monitoring stations. Consequently, further validation is required before conclusions are extended to other spectrum bands, sensing environments, or anomaly categories.

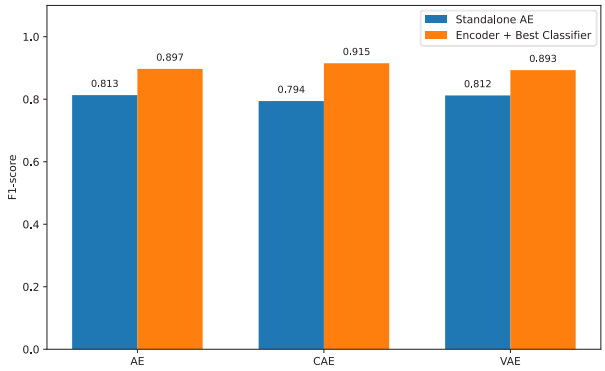


Fig. 4. F1-score comparison between standalone autoencoders and the best encoder–classifier pairs (per encoder).

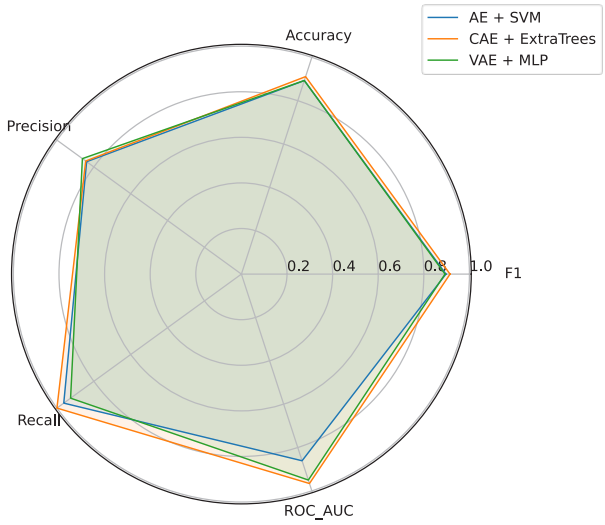


Fig. 5. Radar chart summarizing F1-score, Accuracy, Precision, Recall, and ROC AUC for the best-performing encoder–classifier combinations.

TABLE III
CROSS-VALIDATION F1-SCORES OF SUPERVISED CLASSIFIERS FOR EACH ENCODER FEATURE SPACE.

Classifier	AE	CAE	VAE
RandomForest	0.864	0.826	0.884
GradientBoosting	0.894	0.844	0.930
XGBoost	0.878	0.889	0.930
LogisticRegression	0.857	0.864	0.905
SVM	0.889	0.844	0.930
kNN	0.870	0.851	0.894
GaussianNB	0.884	0.895	0.895
ExtraTrees	0.870	0.889	0.930
MLP	0.909	0.870	0.930
LightGBM	0.844	0.851	0.900

VI. CONCLUSIONS

This study comparatively evaluated three autoencoder architectures—AE, CAE, and VAE—for RF spectrum anomaly detection using FM-band waterfall measurements. Standalone reconstruction-based autoencoders achieved moderate detection performance, whereas substantially stronger results were obtained when encoder-derived latent representations were combined with supervised classifiers.

Among the evaluated configurations, the CAE + ExtraTrees pipeline achieved the best overall performance, suggesting that convolutional feature extraction is particularly well suited to capturing discriminative patterns in waterfall-image data.

These findings indicate that hybrid encoder–classifier approaches may serve as practical decision-support tools for spectrum monitoring environments where labeled anomalies are limited. However, the results are restricted to the investigated FM-band dataset, whose size is relatively limited for deep learning applications, and to the specific measurement conditions considered in this study.

Future work should include broader multi-band datasets, external validation, stronger end-to-end deep learning baselines, and deployment-oriented real-time testing.

VII. DATA AVAILABILITY STATEMENT

The RF spectrum measurement data used in this study were obtained from the Hungarian National Spectrum Monitoring Network, operated under the national regulatory authority. Due to regulatory and confidentiality restrictions, the full raw measurement data cannot be made publicly available.

- A limited set of 20 representative waterfall spectrum images [14] is openly available at Zenodo: S. L. Takács. (2025). Spectrum images in the FM frequency band. Zenodo. https://doi.org/10.5281/zenodo.17174395
- The program codes and Jupyter Notebook implementing the preprocessing and training pipeline, including autoencoder models and supplementary scripts [15], are openly available at Zenodo: S. L. Takács. (2025). Autoencoder Models and Supplementary Code for RF Spectrum Anomaly Detection (FM Band). Zenodo. https://doi.org/10.5281/zenodo.17174452

Additional derived data may be shared upon reasonable request, subject to regulatory approval.

REFERENCES

[1] Hungarian National Assembly, "Act C of 2003 on Electronic Communications," 2003. [Online]. Available: https://net.jogtar.hu/jogszabaly?docid=a0300100.tv. Accessed: Aug. 26, 2025.

[2] S. Rajendran, W. Meert, V. Lenders, and S. Pollin, "Unsupervised wireless spectrum anomaly detection with interpretable features," *IEEE Trans. Cogn. Commun.* Netw., vol. 5, pp. 637–647, 2019. DOI: 10.1109/TCCN.2019.2911524

[3] Q. Cheng, Z. Shi, D. N. Nguyen, and E. Dutkiewicz, "Deep learning network based spectrum sensing methods for OFDM systems," *arXiv preprint arXiv:1807.09414*, 2018. [Online]. Available: https://arxiv.org/abs/1807.09414. Accessed: Aug. 26, 2025.

[4] J. Zhao, L. Hu, Y. Dong, L. Huang, S. Weng, and D. Zhang, "A combination method of stacked autoencoder and 3D deep residual network for hyperspectral image classification," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 102, Art. no. 102 459, 2021. DOI: 10.1016/j.jag.2021.102459

[5] Z. Lin, Y. Chen, X. Zhao, and G. Wang, "Spectral–spatial classification of hyperspectral image using autoencoders," in *Proc. 9th Int. Conf. Inf., Commun. Signal Process. (ICICS)*, Tainan, Taiwan, Dec. 2013, pp. 1–5. DOI: 10.1109/ICICS.2013.6782778

[6] N. Tandiya, A. Jauhar, V. Marojevic, and J. H. Reed, "Deep predictive coding neural network for RF anomaly detection in wireless networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Kansas City, MO, USA, May 2018, pp. 1–6. DOI: 10.1109/ICCW.2018.8403654

[7] S. Ahmad, K. Styp-Rekowski, S. Nedelkoski, and O. Kao, "Autoencoder-based condition monitoring and anomaly detection method for rotating machines," in *Proc. IEEE Int. Conf. Big Data*, Atlanta, GA, USA, Dec. 2020, pp. 4093–4102. DOI: 10.1109/BigData50022.2020.9378015

[8] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, Halifax, NS, Canada, Aug. 2017, pp. 665–674. DOI: 10.1145/3097983.3098052

[9] J. Batson, C. G. Haaf, Y. Kahn, and D. A. Roberts, "Topological obstructions to autoencoding," *J. High Energy Phys.*, vol. 2021, no. 4, pp. 1–43, 2021. DOI: 10.1007/JHEP04(2021)280

[10] L. Beggel, M. Pfeiffer, and B. Bischl, "Robust anomaly detection in deep autoencoders using adversarial training," *Mach. Learn.*, vol. 109, pp. 1611–1630, 2020. DOI: 10.48550/arXiv.1901.06355

[11] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015. [Online]. Available: https://arxiv.org/abs/1412.6572. Accessed: Aug. 26, 2025.

[12] A. Selim, F. Paisana, J. A. Arokkiam, Y. Zhang, L. Doyle, and L. A. DaSilva, "Spectrum monitoring for radar bands using deep convolutional neural networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Singapore, Dec. 2017, pp. 1–6. DOI: 10.1109/GLOCOM.2017.8254105

[13] E. V. Vijay and K. Aparna, "Spectrum sensing using deep learning for proficient data transmission in wireless sensor networks," *Ain Shams Eng. J.*, 2024. DOI: 10.1016/j.asej.2024.102784

[14] S. L. Takács, "Spectrum Images in the FM Frequency Band," *Zenodo*, 2025. DOI: 10.5281/zenodo.17174395

[15] S. L. Takács, "Autoencoder Models and Supplementary Code for RF Spectrum Anomaly Detection (FM Band)," *Zenodo*, 2025. DOI: 10.5281/zenodo.17174452

Szilárd László Takács was born in Szombathely, Hungary, on July 25, 1997. He received the B.Sc. degree in electrical engineering in 2019 and the M.Sc. degree in electrical engineering in 2022 from Széchenyi István University, Győr, Hungary. He also received the M.Sc. degree in physics in 2022 from the Budapest University of Technology and Economics, Budapest, Hungary, and the postgraduate diploma in mathematics expert in data analytics and machine learning in 2023 from Eötvös Loránd University, Budapest, Hungary. He began his Ph.D. studies in 2022 at Széchenyi István University, Győr, Hungary, in the Multidisciplinary Doctoral School of Engineering Sciences, specializing in informatics. His major field of study is artificial intelligence.
He is currently working at the National Media and Infocommunications Authority (NMHH), Hungary, as a Technology Analyst. He also serves as an Assistant Lecturer at Széchenyi István University, Győr, Hungary, where he teaches courses in Microwave Technology I–II, Signal Processing and Quantum Communication. He previously held teaching positions at the Budapest University of Technology and Economics as a Demonstrator and Teaching Assistant in Mathematics between 2021 and 2022, and he was a Research Assistant at the Wigner Research Centre for Physics, Department of Applied and Nonlinear Optics, in the Ultrafast Nanooptics "Lendület" Research Group. His research focuses on the application of artificial intelligence in signal processing. He has published works in this field, including "Applicability of Vanilla Autoencoder and Convolutional Autoencoder in the Field of Radio Spectrum Monitoring," presented at the XXIX International PhD Conference of Professors for the European Hungary Association, Budapest, Hungary, in 2025.
Mr. Takács is a member of several national professional communities in the fields of telecommunications and information technology. His current professional interests include artificial intelligence, data-driven technologies, and signal processing. He is actively involved in academic education and research collaboration projects aimed at integrating machine learning techniques into modern communication systems.

Konrád Kajdy was born in Nagyatád, Hungary in 2004. He is currently studying computer science at Széchenyi István University in Győr, and will soon receive his B.Sc. degree in 2026.
In 2024 he spent a semester abroad in Seoul, South Korea at Chung-Ang University. Currently he is working at NMHH in Budapest, mainly focusing on researching modern technologies.

MTL-BERTNet: A Multi-Task Learning Framework for Aspect and Sentiment Analysis in MOOC Reviews

[1]Raja Ouadad, and Hicham Mouncif

***Abstract*—In the context of large-scale online learning environments, analyzing student feedback is crucial for improving course content and learner engagement. This paper proposes MTL-BERTNet, a novel multi-task learning architecture that jointly performs aspect category classification and sentiment polarity detection from MOOC reviews. The model leverages contextual embeddings from a pre-trained BERT encoder and integrates a convolutional multi-head attention mechanism to capture subtle semantic nuances and inter-task dependencies. To further enhance shared representation learning across tasks, an inter-task matching layer (IML) is introduced. Experiments conducted on an imbalanced MOOC review dataset demonstrate strong performance, with macro F1-scores of 0.90 for aspect classification and 0.93 for sentiment prediction. These results highlight the effectiveness of jointly modeling aspects and sentiment, offering practical insights for improving course design, instructional quality, and learner satisfaction in MOOC platforms.**

***Index Terms*—aspect-based sentiment analysis, multi-task learning, MOOC reviews, BERT, deep learning, attention mechanism.**

I. INTRODUCTION

Massive Open Online Courses (MOOCs) generate abundant user-generated reviews, offering insights into learners' experiences and expectations. Analyzing this feedback helps educators and course designers improve course quality, teaching methods, and engagement [1]. While traditional sentiment analysis identifies positive, negative, or neutral sentiments [2], it often misses aspect-specific details. Aspect-based sentiment analysis (ABSA) addresses this by linking sentiments to particular aspects—such as content, teaching style, or platform usability—providing a more detailed and actionable understanding [3].

Learners' feedback on online courses often addresses specific aspects—such as content quality, instructional clarity, course design, or overall structure—paired with a sentiment (positive, negative, or neutral) toward each aspect [4]. For instance, "The content was comprehensive and up to date" expresses positive sentiment toward the content, whereas "The structure was confusing and lacked coherence" conveys negative sentiment about course structure. This illustrates the close connection between aspect identification and sentiment polarity, which is crucial for fine-grained sentiment analysis in MOOCs [5][6].

Most prior work in sentiment analysis focuses on polarity detection alone [7], often neglecting the interdependence between sentiments and the aspects they reference. In learner reviews, sentiments are typically anchored to specific course dimensions, such as content or instructor performance. For example, "The instructor explained concepts poorly" links negative sentiment directly to the instructor aspect. Modeling such dependencies allows one task to inform the other, motivating multi-task learning approaches that exploit the correlation between aspect recognition and sentiment classification to improve performance in both tasks.

To address this issue, we suggest a unified multi-task learning (MTL) system that manages sentiment analysis and aspect categorization at the same time. The model can benefit from the connections between these tasks by training them together, which frequently produces better overall outcomes.

As a kind of implicit regularization that helps avoid overfitting, MTL's strength lies in its capacity to allow related tasks to guide one another [8]. By finding complementary patterns across tasks, this shared learning process helps the model generalize more successfully [9][10]. The advantages of MTL have been demonstrated in a variety of NLP applications, including sentiment and emotion classification, syntactic parsing, and machine translation [11]. In our work, because aspect evaluation and sentiment polarity are fundamentally coupled, MTL provides a viable strategy to exploit this interaction and improve both tasks simultaneously [12].

Unlike traditional multi-task learning (MTL) methods that divide the model into shared and task-specific parts, our framework introduces a new component called the Inter-Task Matching Layer (IML). Rather than relying solely on shared representations, the proposed IML explicitly models bidirectional interactions between aspect category classification and sentiment polarity detection through cross-task information exchange. This layer is designed to better capture and utilize the interplay between both tasks, enabling the model to learn richer and more discriminative task-specific representations [13]. Furthermore, to boost overall

[1] Raja Ouadad, Hicham Mouncif are affiated with laboratory of Innovation and Mathematics, Applications and Information Technology, Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco. (E-mails: ouadadraja2@gmail.com, h.mouncif@usms.ma) Manuscript received October 23, 2025; revised XX,XX,XXXX

DOI: 10.36244/ICJ.2026.2.8

performance, our approach incorporates transfer learning by using the pre-trained BERT model, which is well-regarded for its ability to understand complex semantic and syntactic relationships within text. BERT was selected as a widely adopted and computationally efficient backbone, allowing us to focus on evaluating the effectiveness of the proposed multi-task architecture rather than the impact of different pre-trained language models. In addition, a hybrid CNN–Multi-Head Attention (CNN–MHA) module is employed to jointly capture local contextual patterns and long-range semantic dependencies within learner reviews.

In summary, the main contributions of this study are as follows:

1. We propose a novel BERT-enhanced Multi-Task Learning framework (MTL-BERTNet) designed to jointly perform sentiment polarity detection and aspect category classification on MOOC reviews.
2. A hybrid architecture utilizing convolutional operations and multi-head attention is employed to simultaneously model local patterns and broader context within learner reviews.
3. An Inter-Task Matching Layer is employed to capture dependencies and interactions between the sentiment analysis and aspect classification tasks.
4. We utilize the powerful contextual representation capabilities of pre-trained BERT to initialize the embedding layer, enabling the model to benefit from rich linguistic features.
5. Extensive experiments conducted on a real-world imbalanced dataset demonstrate that our proposed MTL-BERTNet model significantly outperforms several strong baselines in both sentiment and aspect classification tasks.

The remainder of this paper is structured as follows: Section II provides an overview of related work. Section III details the proposed multi-task learning framework. Section IV describes the experimental setup and presents the results and analysis. Finally, Section V presents the conclusion of the study and highlights potential avenues for future investigation.

II. RELATED WORK

Aspect-based Sentiment Analysis (ABSA), a subfield of Sentiment Analysis (SA), aims to identify sentiments associated with specific aspects of a target entity [14][19]. In MOOCs, ABSA enables the analysis of learners' opinions regarding cours content, instructional quality, course design, and platform usability, providing actionable insights for educational improvement [5] [14]. Existing approaches can be broadly categorized into traditional and deep learning methods, transformer-based models, and multi-task learning frameworks.

A. Traditional and Deep Learning Methods

While sentiment analysis has been widely applied across domains, research on educational ABSA remains relatively limited. Early studies relied on traditional machine learning techniques such as Naive Bayes and SVM for analyzing student feedback, achieving accuracies of 72.8% and 81%, respectively [15]. Subsequent work employed Stanford CoreNLP, OpenNLP, SentiWordNet, and ontology-based approaches for aspect extraction and sentiment classification in educational settings [16], [17].

Deep learning methods further improved performance by learning richer semantic representations. However, most of these approaches rely on sequential architectures (e.g., LSTMs) that may struggle to capture long-range dependencies and contextual relationships as effectively as transformer-based models. Sindhu et al. [18] proposed a two-layer LSTM architecture achieving 91% accuracy in aspect extraction and 93% in sentiment classification. Similarly, [19] reported F1-scores of 80% and 81% for aspect and sentiment prediction, respectively. Kastrati et al. [20] proposed a hybrid Word2Vec–CNN model for Coursera reviews, achieving about 80–82% macro-F1 and outperforming classical baselines by 6–10% in F1-score. A later study [21] used weak supervision and reported an F1-score of approximately 93.3%, showing strong performance while reducing annotation cost.

B. BERT and Transformer-Based Models

More recent transformer architectures have continued to improve contextual representation learning; however, most studies focus on encoder improvements rather than explicitly modeling the interaction between aspect and sentiment prediction tasks. The introduction of BERT [13] and its variants, including SpanBERT [22] and RoBERTa [23], significantly advanced ABSA by providing contextualized word representations. In educational datasets, transformer-based models consistently outperformed conventional machine learning approaches. Fine-tuned IndoBERT achieved F1-scores of 0.897 and 0.882 for aspect extraction and sentiment classification, respectively [24], while multilingual transformer architectures demonstrated promising results for Arabic educational reviews [25].

Further studies confirmed the effectiveness of transformer-based approaches in educational feedback analysis. IndoBERT achieved 92.9% accuracy on university reviews [26], EduBERT reached 89.78% accuracy in sentiment and urgency classification [27], and hierarchical BERT architectures improved instructor evaluation tasks [28][29]. Hybrid approaches combining transformers and traditional classifiers also reported strong results, with RoBERTa achieving 95.5% sentiment classification accuracy [30].

Despite their effectiveness, most transformer-based approaches model aspect identification and sentiment classification independently, limiting their ability to explicitly exploit the strong dependency between these two tasks.

C. Multi-Task Learning for ABSA

Multi-Task Learning (MTL) has emerged as an effective paradigm for ABSA because it enables related tasks to share representations and improve generalization [8]. Attention-based MTL architectures have shown superior performance compared with single-task models by capturing inter-task dependencies more effectively [31], while task-specific

attention mechanisms and cross-stitch networks further improve performance by enabling adaptive feature sharing across tasks, with reported gains on standard multi-task benchmarks such as text and vision classification datasets [32].

He et al. [33] proposed an MTL framework that jointly addresses Aspect Term Extraction (ATE), Aspect Polarity Classification (APC), and aspect–sentiment pair recognition. Zhao et al. [34] incorporated ATE as an auxiliary task and leveraged multi-head attention to strengthen dependency modeling. Other studies combined BERT with knowledge graphs and graph neural networks to enrich semantic representations and improve ABSA performance [35].

More recently, SP-MTL-BERT employed selective paraphrasing with nuance control to augment educational review datasets, reporting substantial gains in aspect recall and sentiment precision [36]. Although SP-MTL-BERT and the proposed framework both address aspect and sentiment analysis in educational reviews within a multi-task learning setting, their design objectives differ. SP-MTL-BERT primarily focuses on data augmentation and shared transformer representations, whereas MTL-BERTNet emphasizes explicit cross-task knowledge exchange through task-specific projections and a co-attention-based Inter-Task Matching Layer (IML). While existing MTL approaches generally rely on shared parameters, auxiliary-task supervision, or attention mechanisms to facilitate knowledge transfer, the dependency between aspect categories and sentiment polarity is often modeled only indirectly. To address this limitation, MTL-BERTNet incorporates the IML together with a hybrid CNN–MHA encoder, enabling both explicit inter-task interaction and the joint modeling of local contextual patterns and long-range semantic dependencies. As a result, the proposed framework provides a more direct mechanism for exploiting the mutual relationship between aspect categories and sentiment polarity.

III. PROPOSED METHOD

We present a multi-task learning framework for joint classification of aspect categories and sentiment polarity in MOOC reviews. Joint modeling captures interdependencies between sentiment and aspect information, improving overall performance in aspect-based sentiment analysis.

A. Formal Task Definition

Let $R = [x_1, x_2, \ldots, x_N]$ represent a learner review composed of N tokens, where each token x_i denotes a word in the input sequence. Each review in the dataset is annotated with two types of labels:

• A sentiment polarity label, denoted by Y_{sent}, belonging to the set Y_{sent}, defined as:

Y_{sent} = {Positive (P), Neutral (NEU), Negative (N)}

• An aspect category label, denoted by Y_{aspect}, belonging to the set Y_{aspect}, defined as:

Y_{aspect} = {Content (C), Structure (S), Instructor (I), Course Design (SC), General Impression (ISC)}

The problem is formulated as a multi-task learning (MTL) framework where both tasks are jointly optimized, enabling the model to exploit semantic relationships between aspect and sentiment through shared contextual representations.

B. Model Architecture Overview

We propose MTL-BERTNet (Multi-Task Learning BERT Network), illustrated in Figure 1. The architecture consists of five main components:

- BERT Embedding Layer: Generates contextualized token embeddings $B_{emb} \in \mathbb{R}^{N \times d}$, where $d = 768$.
- Shared Representation Learning Module: A hybrid CNN–Multi-Head Attention (CNN–MHA) module extracts local and global contextual features, producing the shared representation C.
- Task-Specific Representation Learning Module: Two Fully Connected (FC) layers transform the shared representation into the sentiment representation S_{rep} and aspect representation A_{rep}.
- Inter-Task Matching Layer (IML): Performs bidirectional information exchange through Aspect-to-Sentiment (A2S) and Sentiment-to-Aspect (S2A) attention, enabling interaction between task-specific representations.
- Task-Specific Classification Layer: Uses Softmax classifiers to predict aspect categories and sentiment polarity.

For clarity, B_{emb} denotes BERT embeddings, C the shared feature representation, FC the Fully Connected layer, S_{rep} the sentiment representation, and A_{rep} the aspect representation.

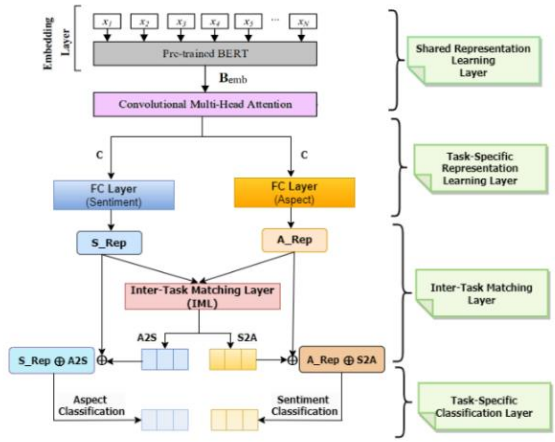


Fig.1. Overall architecture of the proposed Multi-Task Learning BERT Network (MTL-BERTNet), showing the flow from BERT-based embeddings through shared representation learning, task-specific encoding, and inter-task matching via cross-attention, followed by final classification layers for aspect and sentiment prediction.

C. *Layer-Wise Framework and Learning Mechanism*

The first stage of the proposed architecture aims to transform each review into a dense vector representation suitable for downstream learning.

Given an input sequence $R = [x_1, x_2, \dots, x_N]$ of N tokens, a frozen pre-trained BERT model [13] generates contextualized embeddings $B_{\text{emb}} \in \mathbb{R}^{N\times d}$ (with $d = 768$). Unlike static word representations, BERT produces context-dependent embeddings that capture lexical ambiguity and complex syntactic relations. Using BERT as a frozen feature extractor reduces computational cost, mitigates overfitting, and preserves pre-trained linguistic knowledge. This choice is motivated by the relatively limited size of MOOC review datasets and the need to ensure stable contextual representations while reducing training complexity, making it well-suited for our multi-task learning setting.

From this contextual embedding matrix, a hybrid CNN–Multi-Head Attention (CNN–MHA) module extracts high-level semantic features (Figure 2). First, a 1D convolution with kernel size k and ReLU activation captures local dependencies (short phrases, n-grams):

$$X = \text{ReLU}(\text{Conv1D}(B_{\text{emb}}) + b), X \in \mathbb{R}^{N\times d}$$

where $b \in \mathbb{R}^d$ is a learnable bias. To complement these local features, a multi-head self-attention mechanism models global dependencies. Self-attention-based models [37] are effective at this task due to their parallelizable structure, and the multi-head attention mechanism [38] enables learning multiple semantic relationships simultaneously. Let H denote the number of attention heads, with head dimension $d_h = d/H$. For each head $h \in \{1, \dots, H\}$, the local representations X are projected into queries, keys, and values:

$$Q_h = XW_Q^h, K_h = XW_K^h, V_h = XW_V^h$$

The attention operation for head h is:

$$\text{Attention}_h(Q_h, K_h, V_h) = \text{softmax}\left(\frac{Q_h K_h^\top}{\sqrt{d_h}}\right) V_h \in \mathbb{R}^{N\times d_h}$$

where $\frac{Q_h K_h^\top}{\sqrt{d_h}}$ computes scaled pairwise similarities between tokens. The outputs of all heads are concatenated to form a unified representation:

$$C = \text{Concat}(C^1, C^2, \dots, C^H) \in \mathbb{R}^{N\times d},$$

$$C^h = \text{Attention}_h(Q_h, K_h, V_h)$$

Matrix C integrates both local and global features, serving as the shared representation for both tasks.

To specialize this shared knowledge, two linear projections map C into lower-dimensional task-specific spaces ($d' < d$):

$$S_{\text{rep}} = CW_s, A_{\text{rep}} = CW_a$$

with $W_s, W_a \in \mathbb{R}^{d\times d'}$, yielding $S_{\text{rep}}, A_{\text{rep}} \in \mathbb{R}^{N\times d'}$.

A co-attention mechanism then enables bidirectional information exchange between tasks. A normalized similarity matrix is first computed:

$$M = \frac{S_{\text{rep}} A_{\text{rep}}^\top}{\sqrt{d'}} \in \mathbb{R}^{N\times N}$$

Row-wise softmax normalization produces cross-task enriched representations:

$$\text{S2A} = \text{softmax}_{\text{row}}(M^\top) S_{\text{rep}} \in \mathbb{R}^{N\times d'},$$

$$\text{A2S} = \text{softmax}_{\text{row}}(M) A_{\text{rep}} \in \mathbb{R}^{N\times d'}$$

Here, S2A enriches aspect features with sentiment information, while A2S enriches sentiment features with aspect information.

For sentence-level prediction, global average pooling over the sequence length N aggregates token-level representations:

$$F_{\text{sent}} = \text{GlobalAvgPooling}\big(S_{\text{rep}} \oplus \text{A2S}\big) \in \mathbb{R}^{2d'},$$

$$F_{\text{aspect}} = \text{GlobalAvgPooling}(A_{\text{rep}} \oplus \text{S2A}) \in \mathbb{R}^{2d'}$$

where $\oplus$ denotes feature-wise concatenation. Final probability distributions are obtained via softmax classifiers:

$$P_{\text{sent}} = \text{softmax}(F_{\text{sent}} W_{\text{sent}} + b_{\text{sent}}) \in \mathbb{R}^{|Y^{\text{sent}}|},$$

$$P_{\text{aspect}} = \text{softmax}(F_{\text{aspect}} W_{\text{aspect}} + b_{\text{aspect}}) \in \mathbb{R}^{|Y^{\text{aspect}}|}$$

with $|Y^{\text{sent}}| = 3$ and $|Y^{\text{aspect}}| = 5$.

The model is trained via joint optimization, minimizing the sum of categorical cross-entropy losses:

$$\mathcal{L} = \mathcal{L}_{\text{sent}} + \mathcal{L}_{\text{aspect}},$$

$$\mathcal{L}_{task} = -\frac{1}{B}\sum_{n=1}^{B}\sum_{c=1}^{C} y_{n,c} \log(\hat{y}_{n,c})$$

where B is the mini-batch size, C the number of classes, $y_{n,c}$ the one-hot ground truth, and $\hat{y}_{n,c}$ the predicted probability. Equal weighting of both losses ensures balanced learning and effective knowledge transfer between sentiment and aspect classification.

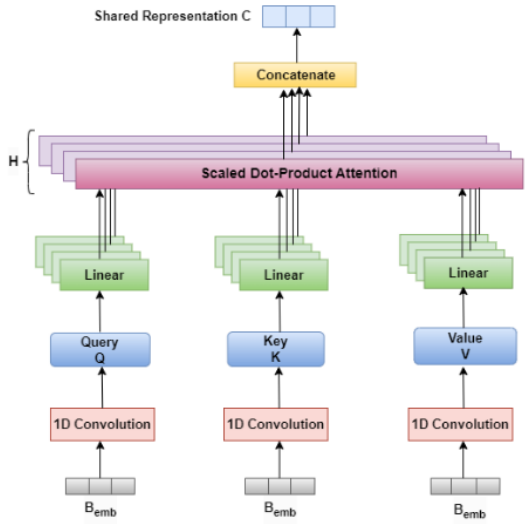


Fig. 2. Architecture of the Convolutional Multi-Head Attention Module (CNN-MHA).

IV . EXPERIMENTS

In this section, we present the experimental setup, dataset details, evaluation metrics, implementation settings, and a comprehensive analysis of the results obtained by the proposed MTL-BERTNet model. The primary objective of this experimental phase is to assess the effectiveness of our multi-task learning approach in jointly performing aspect category classification and sentiment polarity detection on MOOC reviews.

A. Dataset Description

The dataset used in this study consists of 24,255 student reviews collected from the Coursera platform through a web-scraping process. The data spans 15 distinct MOOCs covering multiple subject domains and was collected over a defined time period.

These 24,255 reviews include both authentic and synthetic samples. The dataset is composed of 21,940 (90.5%) authentic reviews and 2,315 (9.5%) synthetic reviews generated using a GPT-based data augmentation strategy.

The authentic reviews were originally collected and manually labeled by Kastrati et al. (2020) [20], and the dataset was obtained directly from the authors upon request, as it is not publicly available online. ensuring high-quality annotations across 15 courses. All reviews are written in English and represent realistic learner feedback from Massive Open Online Courses (MOOCs).

To enhance the representation of less frequent aspect–polarity combinations, synthetic reviews were generated using a GPT-based data augmentation approach. These samples were created using a structured instruction-based prompting strategy designed to produce realistic MOOC-style feedback while maintaining consistency with course-related aspects and sentiment labels. The goal of this augmentation was to increase data diversity and improve model generalization rather than to strictly balance the dataset.

Each instance in the dataset comprises three features: Comment, which contains the textual content of the review; Aspect, referring to one of five predefined course-related categories; and Polarity, indicating the sentiment expressed toward the aspect.

To address class imbalance, class weighting is applied during training, where weights are computed inversely proportional to class frequencies in the training set.

Reviews vary in length from 1 to 554 words, with an average of 23.26 words, a median of 13 words, and a standard deviation of 29.11, reflecting high variability in review length. Detailed descriptions of the Aspect and Polarity categories are provided in Table 1 and Table 2, respectively.

TABLE I
ASPECT CATEGORIES WITH DESCRIPTIONS AND EXAMPLES FROM THE DATASET.

Code	Aspect	Description	Example
I	Instructor	Reviews that focus on instructor-related elements such as teaching style, delivery, communication, or subject expertise.	"funny and engaging instructor breaking down the basics of physics in a simple and easy to understand way!"
C	Content	Feedback concerning the course material, topics covered, assignments, or overall subject matter.	"fantastic, i never took a physics course when i was getting my undergrad degree. this course has been great in helping me to understand "how things work". it was really well put together and easy to understand."
S	Structure	Comments addressing the course's organization, sequencing, or clarity in presenting learning objectives.	"very limited information on wordpress. then added some lectures on web design that could have used information relevant to wordpress such as image compression plugins instead of tinypng.com (see last component)."
SC	Design	Reviews reflecting both the structure and content, emphasizing how they work together to shape the course.	"learning material is too good and very well organised too. overall, i have good learning experience here. thanks coursera."
ISC	General	Broad or overall evaluations that reference multiple aspects of the course, such as content, delivery, and structure.	"everything is awesome .thank you for everything!"

TABLE II
POLARITY CATEGORIES WITH DESCRIPTIONS AND EXAMPLES FROM THE DATASET.

Code	Polarity	Description	Example
N	Negative	Reviews expressing dissatisfaction, criticism, or unfavorable sentiment toward an aspect.	"very bad tool use for uploading assignment."
NEU	Neutral	Comments that are objective, balanced, or without strong positive or negative sentiment.	"you should provide support for python 2 and python 3."
P	Positive	Feedback that conveys satisfaction, praise, or favorable opinions regarding an aspect.	"this course is very useful!"

The distribution of reviews across aspect categories is illustrated in Figure 3. As the figure shows, there is a significant imbalance in the number of reviews assigned to each aspect. For instance, the Content category contains 12,600 reviews, whereas the Structure category includes only 1801 reviews.

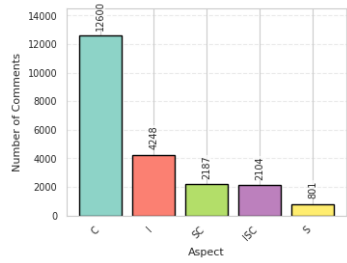


Fig. 3. Distribution of Aspects Mentioned.

The sentiment polarity for each aspect can be classified as positive (P), negative (N), or neutral (NEU). The distribution

of reviews across these polarity categories is presented in Figure 4.

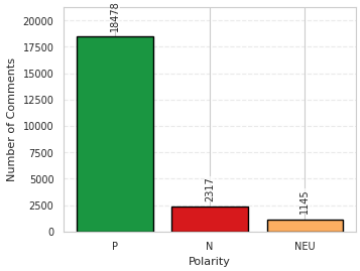


Fig. 4. Distribution of Sentiment Polarities.

B. *Exploratory Analysis of Aspect–Sentiment Interactions*

The exploratory analysis of the MOOC review dataset revealed imbalances between aspects and sentiment polarity. Content (C) is predominantly associated with positive sentiment, whereas Structure (S) and Instructor (I) exhibit a more balanced distribution, including negative and neutral comments. The sentiment–aspect correlation heatmap confirms these patterns, showing high polarity density for positive evaluations in Content and Instructor, while Structure and Design (SC) display a wider sentiment spread.

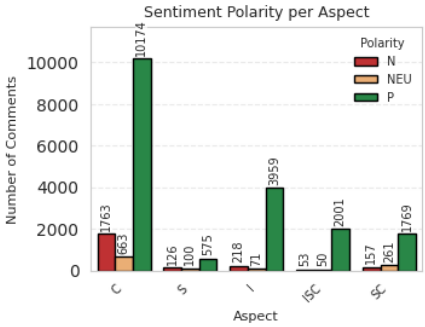


Fig. 5. Sentiment Polarity per Aspect.

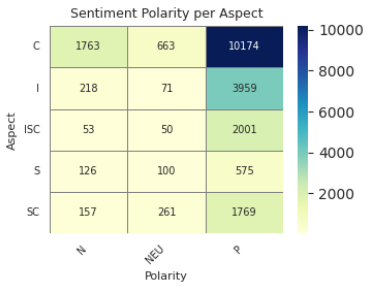


Fig. 6. Sentiment-Aspect correlation heatmap.

Furthermore, the average comment length varies significantly across aspect–sentiment pairs (Figure 7). For instance, negative comments associated with Content and Instructor tend to be substantially longer, potentially reflecting detailed critiques. Conversely, positive comments under General (ISC) or Structure (S) aspects are notably shorter, indicating more concise satisfaction expressions. These insights underscore the nuanced interdependence between sentiment polarity, aspect category, and review verbosity, providing valuable guidance for designing aspect-aware sentiment models capable of capturing such complex patterns.

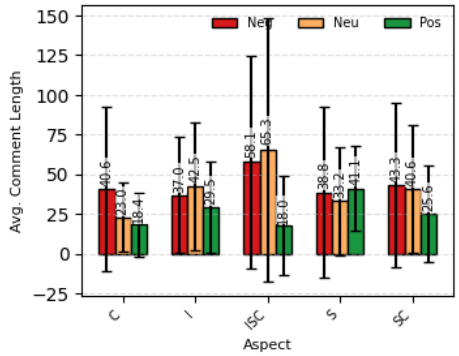


Fig. 7. Average Comment Length by Aspect and Sentiment Polarity.

The exploratory analysis guided key design choices in the proposed model. The imbalance among sentiment–aspect pairs motivated the use of a multi-task learning framework for shared knowledge between aspect and sentiment prediction. Correlations across tasks justified adding an inter-task matching module to enhance semantic representation, while variations in review length and writing style highlighted the need to combine convolutional layers for local context with multi-head attention for long-range dependencies. Together, these insights shaped a robust architecture suited for aspect-based sentiment analysis in MOOCs.

C. *Experimental Setup*

All learner reviews were cleaned by removing HTML tags and extra spaces, then stored in a dedicated column for tokenization. Aspect and polarity labels were converted to integer indices, and the dataset was stratified by aspect–polarity pairs to ensure balanced class representation. Rare combinations with fewer than two samples were discarded. The final split allocated 72% for training, 13% for validation, and 15% for testing. This configuration ensures sufficient training data while maintaining separate validation and test sets for model selection, hyperparameter tuning, and unbiased evaluation.

Reviews were tokenized with the BERT-base-uncased tokenizer and standardized to 128 tokens. Token IDs and attention masks served as model inputs through a custom PyTorch Dataset for multitask learning.

The MTL-BERTNet model employs frozen BERT embeddings followed by a convolutional layer (kernel size = 3, ReLU) and a four-head attention module. Shared features are processed by two task-specific branches linked through inter-task matching layers, enabling mutual learning between aspect and sentiment predictions.

The CNN kernel size of 3 is selected to capture local trigram-level contextual patterns, which is commonly effective for short-text classification tasks such as user reviews. The four-head attention mechanism is used as a lightweight configuration to balance representational capacity and computational efficiency compared to full multi-head settings in standard Transformer architectures.

Training used the AdamW optimizer (learning rate = 2e-5, batch size = 32) for up to ten epochs with early stopping (patience = 2) based on validation macro F1. The learning rate is chosen following standard practice for fine-tuning transformer-based models and is validated using the validation set performance. Class weights were applied to mitigate imbalance in aspect prediction. Experiments were run on GPU with fixed seeds for reproducibility. Evaluation metrics included accuracy, macro F1-score, confusion matrices, and ROC curves. Model parameters are summarized in Table 3.

TABLE III
MODEL AND TRAINING CONFIGURATION.

Component	Value
Tokenizer	BERT-base-uncased
Max Input Length	128 tokens
Embedding Source	Frozen BERT
CNN Kernel Size	3
CNN Activation	ReLU
Attention Heads	4
Inter-Task Matching	MLP (ReLU)
Optimizer	AdamW
Learning Rate	2e-5
Batch Size	32
Epochs	10 (with early stopping)
Early Stopping Patience	2
Loss Function (Aspect)	Cross-Entropy with class weights
Loss Function (Polarity)	Cross-Entropy
Stratified Splitting	By Aspect–Polarity combination
Evaluation Metrics	Accuracy, F1 (macro), ROC, CM
Reproducibility Settings	Random seed = 42
Device	GPU

V. RESULTS AND DISCUSSION

To evaluate the effectiveness of our proposed MTL-BERTNet architecture, we conducted a thorough training procedure across 10 epochs using stratified train–validation–test splits. This stratification preserved the joint distribution of aspect–sentiment label pairs, which is particularly important given the inherent class imbalance in MOOC reviews. To mitigate overfitting, we applied early stopping based on the macro-averaged F1-score of the aspect classification task. The evolution of training and validation metrics—including loss, accuracy, and F1-score—for both tasks is presented in Figure 8.

As shown in Figure 9, the model achieved rapid improvements during the first few epochs, with performance peaking at Epoch 3. Beyond this point, marginal gains in validation F1-score were observed, leading to early stopping after epoch 5. This pattern indicates strong generalization and mitigated overfitting.

The final selected model (epoch 3) attained a validation accuracy of 90.88% for aspect prediction and 96.31% for polarity prediction, along with macro F1-scores of 0.8911 and 0.9403, respectively. These results demonstrate the model's capacity to effectively capture both structural and sentiment-specific information from MOOC reviews.

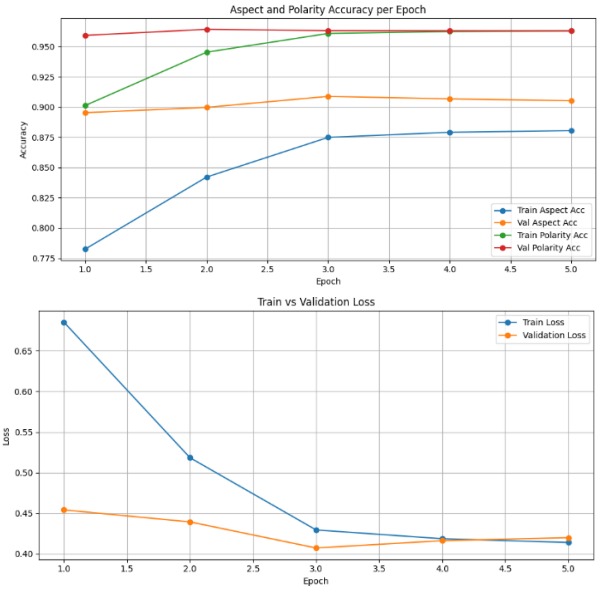


Fig. 8. Model Accuracy and Loss Trends During Training.

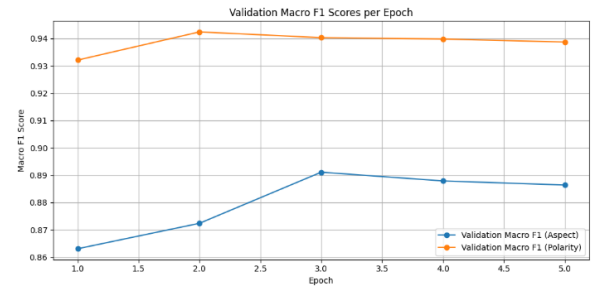


Fig. 9. Validation Macro F1 Score Progression.

To further evaluate generalization, the best model was tested on the held-out test set. The classification performance is summarized in Table 4. The model achieved:

- 92% overall accuracy on aspect classification with a macro F1-score of 0.90, and
- 96% accuracy on polarity classification with a macro F1-score of 0.93.

Notably, the model exhibited strong performance on dominant aspect classes like Content (C) and SC, achieving F1-scores above 0.94. Performance was comparatively lower for underrepresented or semantically diffuse classes such as Structure (S) and Instructor (I). In the sentiment task, the model delivered exceptional performance on Positive (P) comments (F1 = 0.98) and remained robust on more ambiguous classes like Neutral (NEU) and Negative (N).

TABLE IV
TEST SET CLASSIFICATION REPORT.

Aspect Class	Precision	Recall	F1-Score	Support
Content (C)	0.93	0.95	0.94	1891
Instructor (I)	0.85	0.85	0.85	652
General(ISC)	0.95	0.90	0.92	331
Structure (S)	0.83	0.77	0.80	270
Design(SC)	0.96	0.97	0.97	495
Macro Avg	0.91	0.89	0.90	3639

Polarity Class	Precision	Recall	F1-Score	Support
Negative (N)	0.86	0.89	0.88	516
Neutral (NEU)	0.96	0.91	0.93	352
Positive (P)	0.98	0.98	0.98	2771
Macro Avg	0.93	0.93	0.93	3639

The confusion matrices in Figure 10 illustrate MTL BERTNet's performance on the test set, highlighting challenges related to class imbalance. For aspect classification, the model accurately identifies high-resource categories such as Content (C) and SC, with 1792/1891 and 479/495 correct predictions, respectively. Misclassifications occur mainly between Content and Instructor (74 instances) and between Structure and Instructor (11 instances), while 42 Structure samples were predicted as Content, reflecting semantic overlaps between instructional content and pedagogical structure. The underrepresented Instructor class was frequently confused with Content (78) and Structure (8), indicating the difficulty of modeling sparsely represented or diffuse concepts. Despite these issues, recall and precision remain high across all aspect classes.

For polarity prediction, Positive (P) samples were classified with high accuracy (2708/2771), while some confusion exists between Neutral (NEU) and Negative (N). Specifically, 47 Negative samples were predicted as Positive and 10 as Neutral, and 14 Neutral instances were classified as Positive, highlighting the subjective nature of sentiment expression.

These observations underscore the benefit of joint learning in MTL-BERTNet, as shared representations capture interdependencies between aspects and sentiment. For example, structural criticism often conveys subtle negativity, and instructor praise implies positivity even with minimal sentimental cues, helping the model handle nuanced sentiment aspect interactions.

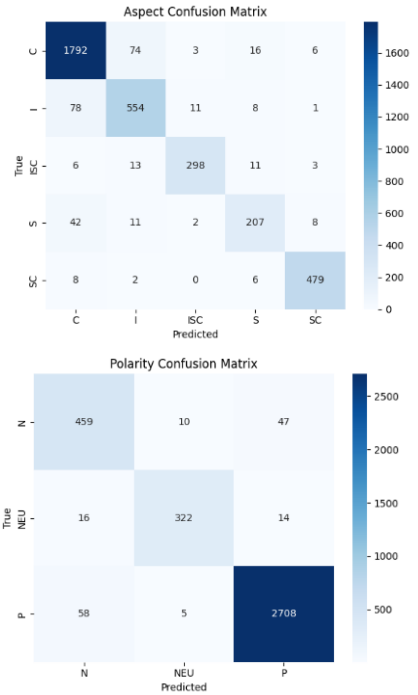


Fig. 10. Confusion Matrices for Aspect and Polarity Tasks.

To gain deeper understanding of how the model handles minority aspect classes, we visualized a filtered confusion matrix for Structure (S) and ISC, two of the least frequent categories, as shown in Figure 11. Despite their low representation, the model correctly identified 196 out of 197 Structure samples and 304 out of 311 ISC samples, with only one Structure sample misclassified and seven ISC samples predicted as Structure. This impressive performance demonstrates that the model's attention and encoding layers are effective at capturing subtle context features, even when class frequencies are low. It also highlights the impact of macro-F1 optimization and stratified sampling during training, which preserve balance and encourage generalization across classes of varying sizes.

Collectively, these findings indicate that MTL-BERTNet is capable of learning both dominant and minority patterns in imbalanced sentiment–aspect datasets. Nonetheless, the remaining misclassifications underscore the need for further refinement, including context-aware attention, data augmentation techniques for rare classes, and integration of external knowledge resources to aid in disambiguating closely related concepts.

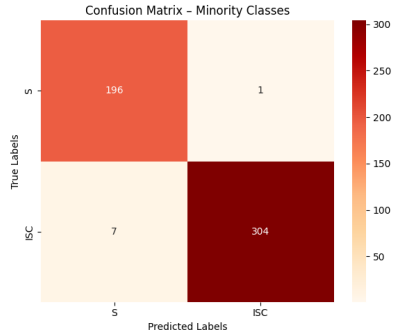


Fig. 11. Filtered Confusion Matrix for Minority Aspect Classes.

To further assess the discriminative capacity of the proposed MTL-BERTNet model, we conducted a multi-class Receiver Operating Characteristic (ROC) analysis for both aspect and polarity classification tasks, as illustrated in Figure 12. These curves provide a comprehensive view of the model's performance across all classes beyond simple accuracy or F1 metrics, particularly in the presence of class imbalance.

As shown in the Figure, the model achieved a macro-average AUC of 0.983 and a micro-averaged AUC of 0.990 for aspect prediction. The class-specific curves highlight that the model performs particularly well on dominant and structurally distinct categories such as SC (AUC = 0.999) and ISC (AUC = 0.994). Even for more semantically overlapping categories like Content (C) and Instructor (I), the AUCs remain strong at 0.981 and 0.976, respectively, indicating a high true positive rate across thresholds.

In the polarity classification task, the ROC analysis further underscores the robustness of the model. With a macro-average AUC of 0.992 and micro-average AUC of 0.996, the model exhibits outstanding predictive performance. Class-specific

AUC values are consistently high: 0.988 for Negative (N), 0.995 for Neutral (NEU), and 0.992 for Positive (P). This high separability suggests that the learned representations effectively capture the sentiment polarity nuances present in learner reviews.

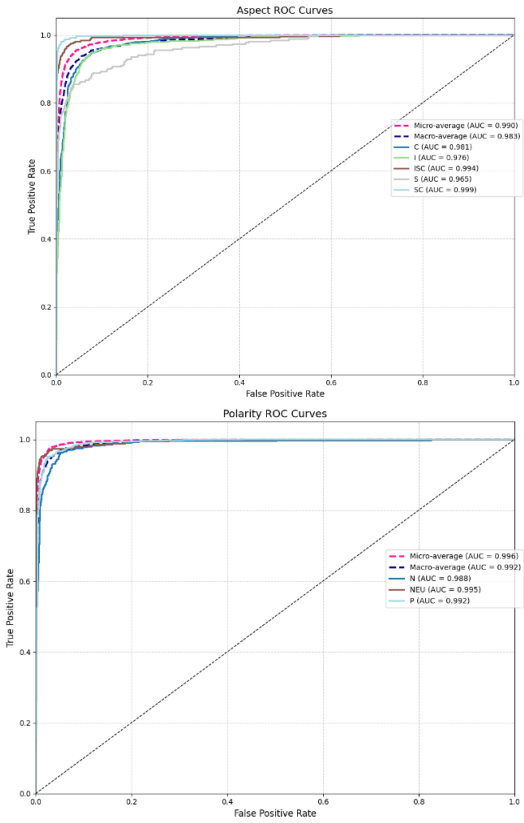


Fig. 12. Multi-Class ROC Curves for Aspect and Polarity Classification (MTL-BERTNet).

These results underscore the effectiveness of our model in capturing the intricate interdependence between aspect classification and sentiment polarity, particularly within the nuanced context of educational reviews. The integration of convolutional filters and multi-head attention mechanisms enhances the model's ability to detect fine-grained linguistic patterns and emphasize contextually salient tokens. Moreover, the inter-task matching layer facilitates cross-task interaction, enabling the model to associate sentiment expressions with their corresponding aspects more coherently. This synergy proves especially valuable in domains like online learning, where user feedback is often dense with implicit opinions and overlapping semantic cues.

From a practical standpoint, these findings have significant implications for the analysis and personalization of MOOCs. Accurate identification of both what learners are commenting on (e.g., course structure, content quality, instructor effectiveness) and how they feel about it (positive, neutral, or negative sentiment) enables more targeted course refinement and improved learner support. For example, consistently low sentiment linked to a specific aspect like "Structure" can alert educators to redesign the learning flow, while strong positive feedback on "Content" can reinforce current pedagogical strategies. Ultimately, by leveraging models like MTL-BERTNet, MOOC platforms can transform unstructured learner feedback into actionable insights, enhancing course quality, increasing engagement, and fostering better learning outcomes.

A. Comparison with Baseline Models

We evaluated the proposed MTL-BERTNet model against several baselines, including classical machine learning algorithms (SVM, Logistic Regression, Random Forest) trained on TF-IDF feature representations of the input text, and deep learning models (CNN, LSTM, BiLSTM). For traditional models, each review was first preprocessed (tokenization, lowercasing, stop-word removal), and then transformed into a TF-IDF weighted bag-of-words vector representation capturing unigram and bigram statistics. Classical methods showed moderate performance for aspect classification (macro F1: 0.51–0.58), with SVM providing the best trade-off between accuracy and generalization. Deep learning models improved sentiment prediction (e.g., LSTM: 88.56% accuracy, 0.77 macro F1) but struggled with less frequent aspect classes, resulting in macro F1 below 0.70.

In contrast, MTL-BERTNet, which combines shared representations, convolutional encoding, and multi-head attention, outperformed all baselines, achieving 92.00% aspect accuracy (macro F1 = 0.90) and 96.00% sentiment accuracy (macro F1 = 0.93). These results demonstrate the effectiveness of multi-task learning in capturing dependencies between aspects and sentiments in MOOC reviews.

TABLE V
PERFORMANCE COMPARISON OF BASELINE MODELS VS. MTL-BERTNET.

Model	Aspect		Sentiment	
	Accuracy	Macro F1	Accuracy	Macro F1
Decision Tree	61.61%	52%	79.61%	61%
Naive Bayes	65.73%	46%	84.53%	64%
SVM	71.56%	58%	87.14%	71%
Random Forest	70.71%	58%	85.68%	68%
Logistic Regression	69.17%	53%	87.11%	72%
CNN	72.31%	64.83%	87.35%	74.19%
LSTM	74.13%	67.94%	88.56%	77.22%
BiLSTM	73.33%	67.65%	87.93%	75.93%
Our Model	92.00%	90%	96.00%	93%

B. Ablation Study

To assess the contribution of each component within our proposed MTL-BERTNet architecture, we conducted a comprehensive ablation study by evaluating several model variants. The BertSTL model serves as a single-task baseline using BERT, applied separately to either aspect or sentiment classification. We tested the Vanilla-MTL-BERTNet version, which removes the Inter-Task Matching Layer (IML), to evaluate the contribution of cross-task knowledge sharing. The MTL-BERTNet-CNN variant excludes the Multi-Head Attention module to assess the importance of capturing global

contextual dependencies, while MTL-BERTNet-NonBERT replaces BERT with pre-trained GloVe embeddings to examine the impact of contextualized representations.

The ablation study is designed to isolate the contribution of each architectural component rather than to maximize absolute performance improvements, which is consistent with standard practice in multi-task learning evaluation.

Compared with the BertSTL baseline, the Vanilla-MTL-BERTNet model improves the Aspect Macro F1-score from 86.10% to 88.76% (+2.66%), demonstrating the effectiveness of the multi-task learning framework. Incorporating the CNN–MHA module further increases the Macro F1-score to 89.42% (+0.66%), highlighting the benefit of jointly capturing local and global contextual dependencies. Adding the Inter-Task Matching Layer yields a further improvement to 90.00% (+0.58%), indicating that cross-task information exchange between aspect and sentiment representations provides complementary knowledge that enhances prediction performance. Overall, these results demonstrate that each component contributes positively and incrementally to the effectiveness of MTL-BERTNet, with the complete model achieving the highest accuracy and Macro F1-score across both tasks.

TABLE VI
ABLATION STUDY RESULTS ON ASPECT AND SENTIMENT CLASSIFICATION.

Model	Aspect		Sentiment	
	Accuracy	Macro F1	Accuracy	Macro F1
BertSTL (Single-task BERT)	88.52%	86.10%	92.14%	90.78%
Vanilla-MTL-BERTNet	90.34%	88.76%	93.85%	92.01%
MTL-BERTNet-CNN	91.21%	89.42%	94.02%	92.38%
MTL-BERTNet-NonBERT (GloVe)	89.65%	87.32%	92.77%	91.24%
MTL-BERTNet (Our Model)	92.00%	90%	96.00%	93%

VI. CONCLUSION

In this work, we introduced MTL-BERTNet, a multi-task learning model that uses BERT to handle aspect classification and sentiment analysis on MOOC reviews simultaneously. Our model captures greater semantic linkages in learner feedback by including an inter-task matching layer and exchanging learnt representations across tasks. The effectiveness and dependability of MTL-BERTNet are demonstrated by experimental results, which routinely outperform single-task techniques and several top baseline models.

These results demonstrate how useful multi-task learning frameworks are for better understanding the thoughts and concerns of MOOC participants, which can support course improvement and more personalized learning experiences.

Future work will evaluate MTL-BERTNet on larger and more diverse educational datasets across multiple MOOC platforms to further assess its robustness and generalization capability. We also plan to integrate explainability techniques to provide more interpretable and actionable insights for educators and platform developers. In addition, exploring domain adaptation methods and incorporating real-time feedback represent promising directions to enhance the model's practicality and real-world applicability.

REFERENCES

[1] R. Ouadad and H. Mouncif, "Sentiment Analysis of Students Feedback in Online Courses Using Supervised, Ensemble, and Transfer Learning Methods," in *Proceedings of the 7th International Conference on Networking, Intelligent Systems and Security*, in NISS '24. New York, NY, USA: Association for Computing Machinery, Aug. 2024, pp. 1–7. DOI: 10.1145/3659677.3659733.

[2] B. Liu, "Sentiment Analysis: A Fascinating Problem," in Sentiment Analysis and Opinion Mining, B. Liu, Ed., Cham: Springer International Publishing, 2012, pp. 1–8. DOI: 10.1007/978-3-031-02145-9_1.

[3] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "SemEval-2014 Task 4: Aspect Based Sentiment Analysis," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, P. Nakov and T. Zesch, Eds., Dublin, Ireland: Association for Computational Linguistics, Aug. 2014, pp. 27–35. DOI: 10.3115/v1/S14-2004.

[4] J. Zeng, K. Luo, Y. Lu, and M. Wang, "An Evaluation Framework for Online Courses Based on Sentiment Analysis Using Machine Learning," *Int. J. Emerg. Technol. Learn.* IJET, vol. 18, pp. 4–22, Sep. 2023, DOI: 10.3991/ijet.v18i18.42521.

[5] W. Awadh, R. Sulaiman, and M. Mahmoud, "Aspect-based sentiment analysis in MOOCs: a systematic literature review introducing the MASC- MEF framework," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 37, Mar. 2025, DOI: 10.1007/s44443-025-00018-1.

[6] S. Deshpande *et al.*,"Elevating educational insights: sentiment analysis of faculty feedback using advanced machine learning models," *Adv. Contin. Discrete Models*, vol. 2025, Apr. 2025, DOI: 10.1186/s13662-025-03933-9.

[7] O. Habimana, Y. Li, R. Li, X. Gu, and G. Yu, "Sentiment analysis using deep learning approaches: an overview," *Sci. CHINA Inf. Sci.*, vol. 63, no. 1, Dec. 2019, DOI: 10.1007/S11432-018-9941-6.

[8] S. Ruder, "An Overview of Multi-Task Learning in Deep Neural Networks," Jun. 15, 2017, *arXiv: arXiv:1706.05098*. DOI: 10.48550/arXiv.1706.05098.

[9] H. Zhang, L. Xiao, W. Chen, Y. Wang, and Y. Jin, "Multi-Task Label Embedding for Text Classification," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds., Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 4545–4553. DOI: 10.18653/v1/D18-1484.

[10] R. Caruana, "Multitask Learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997, DOI: 10.1023/A:1007379606734.

[11] A. Maurer, M. Pontil, and B. Romera-Paredes, "The Benefit of Multitask Representation Learning".

[12] G. Balikas, S. Moura, and M.-R. Amini, "Multitask Learning for Fine- Grained Twitter Sentiment Analysis," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, in SIGIR '17. New York, NY, USA: Association for Computing Machinery, Aug. 2017, pp. 1005–1008. DOI: 10.1145/3077136.3080702.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*: Human Language Technologies, Volume 1 (Long and Short Papers), J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[14] C. Abdessamad and N. El Faddouli, "Sentiment Analysis on Massive Open Online Courses (MOOCs): Multi-Factor Analysis, and Machine Learning Approach," *Int. J. Inf. Commun. Technol. Educ.*, vol. 18, pp. 1–22, Jan. 2022, DOI: 10.4018/IJICTE.310004.

[15] N. D. Valakunde and M. S. Patwardhan, "Multi-aspect and Multi-class Based Document Sentiment Analysis of Educational Data Catering Accreditation Process," in *2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, Nov. 2013, pp. 188–192. DOI: 10.1109/CUBE.2013.42.

[16] L. Balachandran and A. Kirupananda, "Online reviews evaluation system for higher education institution: An aspect based sentiment analysis tool," in *2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, Dec. 2017, pp. 1–7. DOI: 10.1109/SKIMA.2017.8294118.

[17] N. Soe and P. T. Soe, "Domain Oriented Aspect Detection for Student Feedback System," pp. 90–95, Nov. 2019, DOI: 10.1109/aitc.2019.8921372.

[18] I. Sindhu, S. M. Daudpota, K. Badar, M. Bakhtyar, J. Baber, and M. Nurunnabi, "Aspect-based opinion mining on student's feedback for faculty teaching performance evaluation," *IEEE Access*, vol. 7, pp. 108 729–108 741, 2019.

[19] G. S. Chauhan, P. Agrawal, and Y. K. Meena, "Aspect-Based Sentiment Analysis of Students' Feedback to Improve Teaching–Learning Process," in *Information and Communication Technology for Intelligent Systems,* Springer, Singapore, 2019, pp. 259–266. DOI: 10.1007/978-981-13-1747-7_25.

[20] Z. Kastrati, B. Arifaj, A. Lubishtani, F. Gashi, and E. Nishliu, "Aspect-Based Opinion Mining of Students' Reviews on Online Courses," in *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence*, in ACM Other conferences., 2020, pp. 510–514. DOI: 10.1145/3404555.3404633.

[21] Z. Kastrati, A. S. Imran, and A. Kurti, "Weakly Supervised Framework for Aspect-Based Sentiment Analysis on Students' Reviews of MOOCs," *IEEE Access*, vol. 8, pp. 106 799–106 810, 2020, DOI: 10.1109/ACCESS.2020.3000739.

[22] M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, and O. Levy, "SpanBERT: Improving Pre-training by Representing and Predicting Spans," Jan. 18, 2020, *arXiv: arXiv:1907.10529*. DOI: 10.48550/arXiv.1907.10529.

[23] Y. Liu *et al*., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," Jul. 26, 2019, *arXiv: arXiv:1907.11692*. DOI: 10.48550/arXiv.1907.11692.

[24] A. Jazuli, Widowati, and R. Kusumaningrum, "Aspect-based sentiment analysis on student reviews using the Indo-Bert base model," presented at the E3S Web of Conferences, Aug. 2023, p. 02004. DOI: 10.1051/e3sconf/202344802004.

[25] K. Alshaikh, O. Almatrafi, and Y. Abushark, "BERT-based Model for Aspect-Based Sentiment Analysis for Analyzing Arabic Open-ended Survey Responses: A Case Study," *IEEE Access,* vol. PP, pp. 1–1, Jan. 2023, DOI: 10.1109/ACCESS.2023.3348342.

[26] I. Id, H. Saputra, S. Syamsudhuha, R. Kurniawan, and Y. Andriyani, "Sentiment analysis of student evaluation feedback using transformer-based language models," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 36, p. 1127, Nov. 2024, DOI: 10.11591/ijeecs.v36.i2.pp1127-1139.

[27] B. Clavié and K. Gal, "EduBERT: Pretrained Deep Language Models for Learning Analytics," Dec. 02, 2019, *arXiv: arXiv:1912.00690*. DOI: 10.48550/arXiv.1912.00690.

[28] W. Wang, H. Zhuang, M. Zhou, H. Liu, and B. Li, "What Makes a Star Teacher? A Hierarchical BERT Model for Evaluating Teacher's Performance in Online Education," Dec. 03, 2020, *arXiv: arXiv:2012.01633*. DOI: 10.48550/arXiv.2012.01633.

[29] M. Herath, K. Chamindu, H. Maduwantha, and S. Ranathunga, "Dataset and Baseline for Automatic Student Feedback Analysis," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, and S. Piperidis, Eds., Marseille, France: European Language Resources Association, Jun. 2022, pp. 2042–2049. Accessed: Jul. 21, 2025. [Online]. Available: https://aclanthology.org/2022.lrec-1.219/

[30] A. Koufakou, "Deep learning for opinion mining and topic classification of course reviews," *Educ. Inf. Technol.*, vol. 29, no. 3, pp. 2973–2997, Feb. 2024, DOI: 10.1007/s10639-023-11736-2.

[31] X. Li and W. Lam, "Deep Multi-Task Learning for Aspect Term Extraction with Memory Interaction," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, M. Palmer, R. Hwa, and S. Riedel, Eds., Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 2886–2892. DOI: 10.18653/v1/D17-1310.

[32] P. Liu, X. Qiu, and X. Huang, "Adversarial Multi-task Learning for Text Classification," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers), R. Barzilay and M.-Y. Kan, Eds., Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1–10. DOI: 10.18653/v1/P17-1001.

[33] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An Interactive Multi-Task Learning Network for End-to-End Aspect-Based Sentiment Analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Màrquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 504–515. DOI: 10.18653/v1/P19-1048.

[34] G. Zhao, Y. Luo, Q. Chen, and X. Qian, "Aspect-based sentiment analysis via multitask learning for online reviews," *Knowl.-Based Syst.*, vol. 264, p. 110 326, Mar. 2023, DOI: 10.1016/j.knosys.2023.110326.

[35] Z. He, H. Wang, and X. Zhang, "Multi-Task Learning Model Based on BERT and Knowledge Graph for Aspect-Based Sentiment Analysis," *Electronics*, vol. 12, no. 3, Art. no. 3, Jan. 2023, DOI: 10.3390/electronics12030737.

[36] S. Gul, M. Asif, F.-E.- Amin, K. Saleem, and M. Imran, "Advancing Aspect-Based Sentiment Analysis in Course Evaluation: A Multi-Task Learning Framework With Selective Paraphrasing," *IEEE Access,* vol. 13, pp. 7764–7779, 2025, DOI: 10.1109/ACCESS.2025.3527367.

[37] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan, and C. Zhang, "DiSAN: Directional Self-Attention Network for RNN/CNN-Free Language Understanding," Nov. 20, 2017, *arXiv: arXiv:1709.04696*. DOI: 10.48550/arXiv.1709.04696.

[38] A. Vaswani *et al*., "Attention Is All You Need," Aug. 02, 2023, *arXiv: arXiv:1706.03762*. DOI: 10.48550/arXiv.1706.03762.

[39] A. W. Yu *et al*., "QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension," Apr. 23, 2018, *arXiv: arXiv:1804.09541*. DOI: 10.48550/arXiv.1804.09541.

[40] B. Yang, L. Wang, D. Wong, L. S. Chao, and Z. Tu, "Convolutional Self-Attention Networks," Apr. 05, 2019, *arXiv: arXiv:1904.03107*. DOI: 10.48550/arXiv.1904.03107.

[41] N. Majumder, S. Poria, H. Peng, N. Chhaya, E. Cambria, and A. Gelbukh, "Sentiment and Sarcasm Classification With Multitask Learning," *IEEE Intell. Syst.*, vol. 34, no. 3, pp. 38–43, May 2019, DOI: 10.1109/MIS.2019.2904691.

[42] C. Xiong, V. Zhong, and R. Socher, "Dynamic Coattention Networks For Question Answering," Mar. 06, 2018, *arXiv: arXiv:1611.01604*. DOI: 10.48550/arXiv.1611.01604.

Raja Ouadad received the M.Sc. degree in Computer Science from the Faculty of Sciences and Techniques, Beni Mellal, Morocco, in 2022. Currently, she is a Ph.D. candidate in the Department of Mathematics and Informatics at the Polydisciplinary Faculty, Sultan Moulay Slimane University, Morocco. Her research interests include sentiment analysis, artificial intelligence, machine learning, and deep learning algorithms. She is the corresponding author of this article and can be contacted at e-mail: ouadadraja2@gmail.com.

Hicham Mouncif is a Full Professor and Ph.D. Supervisor in the Department of Computer Sciences, Polydisciplinary Faculty of Beni Mellal, University Sultan Moulay Slimane. He has published numerous academic papers in distinguished journals based on teaching and research experience. As the Coordinator of the Computer Systems Engineering Master's Program, his research interests include educational technologies, machine learning, transportation networking, and routing protocols. He can be contacted at e-mail: h.mouncif@usms.ma.

YOLOv8-Based Firearm Detection for Real-Time Video Surveillance Applications

Rexhep Mustafovski, Tomislav Shuminoski, *Member, IEEE*, and Aleksandar Risteski, *Senior Member, IEEE*

***Abstract*—Real-time firearms detection is key for ensuring safety in public spaces and security-sensitive environments. In this paper, we present a computer vision approach using the YOLOv8 object detection model to identify whether an individual is carrying a firearm. The model's architecture allows for rapid and accurate detection, making it suitable for real-time applications. A custom dataset of images was used for training and validation, with preprocessing techniques applied to enhance model generalization. The proposed system was evaluated on key performance metrics, achieving high precision and recall in detecting both exposed and partially concealed firearms. The ability to operate in real-time offers a promising solution for improving security measures in various contexts. Future work will focus on refining the system to reduce false positives and expanding the dataset to include a broader range of scenarios.**

***Index Terms*—AI, Computer Vision, Firearms Detection, YOLOv8**

I. INTRODUCTION

In military operations, distinguishing between combatants and non-combatants is a critical task that can have significant operational and ethical implications. Identifying whether an individual is carrying a firearm or not is one of the primary means of differentiating armed combatants from unarmed civilians in conflict zones. However, manual identification can be highly error-prone, especially in dynamic environments with large crowds or concealed weapons. Misidentification can lead to catastrophic consequences, including civilian casualties, operational setbacks, and violations of international humanitarian law. Automated systems leveraging computer vision and deep learning have the potential to significantly enhance situational awareness and support rapid, data-driven decision-making in such environments [1].

Rexhep Mustafovski is with the Faculty of Electrical Engineering and Information Technologies, University "St. Cyril and Methodius" – Skopje, Skopje, Republic of North Macedonia
(e-mail: redzep.mustafovski@ugd.edu.mk)

Tomislav Shuminoski (Member IEEE) is with the Faculty of Electrical Engineering and Information Technologies, University "St. Cyril and Methodius" – Skopje, Skopje, Republic of North Macedonia (e-mail: tomish@feit.ukim.edu.mk)

Aleksandar Risteski (Senior Member IEEE) is with the Faculty of Electrical Engineering and Information Technologies, University "St. Cyril and Methodius" – Skopje, Skopje, Republic of North Macedonia
(e-mail: acerist@feit.ukim.edu.mk)

Object detection models, in particular, have demonstrated remarkable success in identifying various types of objects within images or video streams in real-time. Among these models, the You Only Look Once (YOLO) family of detectors has gained prominence due to its ability to balance detection speed and accuracy, making it highly suitable for military applications where split-second decisions are required. Despite advances in deep learning, the task of firearm detection in military zones remains a challenging problem. Factors such as poor visibility, cluttered backgrounds, and the concealment of weapons can hinder the performance of existing models. Moreover, there is a limited amount of research focused specifically on real-time firearm detection in combat scenarios, particularly in distinguishing combatants from non-combatants [2]. The objective of this research is to address these gaps by developing a YOLOv8-based system capable of recognizing firearms with high accuracy, providing military personnel with a reliable tool for decision-making in high-stakes environments.

Recent conflicts and asymmetric warfare scenarios have highlighted the need for precise identification systems capable of minimizing collateral damage. While artificial intelligence (AI) and deep learning have demonstrated strong performance in tasks such as surveillance, target recognition, and situational awareness, the application of these techniques specifically for distinguishing combatants from non-combatants based on firearm presence remains limited.

Existing studies often rely on constrained datasets or do not address real-time performance in complex operational environments characterized by occlusions, dynamic movement, and heterogeneous backgrounds. This limitation represents a significant research gap in the deployment of reliable detection systems in military contexts.

The YOLO series of object detection models is well-known for its speed and accuracy in detecting various objects, including vehicles, weapons, and even specific behaviors. YOLOv8, a widely adopted version of the YOLO family, builds on these strengths by providing an optimized architecture that balances detection speed and accuracy, making it suitable for real-time object detection applications.

This makes it particularly suitable for military applications, where delays in detection can have severe consequences. Unlike earlier versions, YOLOv8 incorporates innovations that improve detection in cluttered environments, allowing for more reliable recognition of partially concealed or obscured firearms.

DOI: 10.36244/ICJ.2026.2.9

These improvements are particularly important in urban and densely populated environments, where civilians and combatants are often in close proximity [3].

The proposed system leverages a custom dataset designed to reflect real-world operational conditions in military environments, including variations in lighting, crowd density, and the presence of non-combatant objects such as bags or tools. The system is evaluated using multiple performance metrics, with the objective of providing a reliable decision-support tool for military personnel, enhancing operational effectiveness while supporting compliance with international humanitarian law [1–7].

The objective of this study is to develop and evaluate a YOLOv8-based object detection system capable of accurately identifying firearms in real-time under diverse environmental conditions. The proposed approach focuses on improving detection performance and supporting decision-making processes in security and surveillance scenarios. Although this work does not introduce a new detection architecture, its contribution lies in the application and evaluation of a YOLOv8-based system for firearm detection in operationally relevant scenarios, supported by a curated dataset and system-level integration.

The main research question addressed in this study is whether a YOLOv8-based detection system can achieve reliable and accurate firearm detection in real-time under diverse and challenging environmental conditions.

II. LITERATURE REVIEW

Recent studies in object detection have demonstrated significant improvements in both accuracy and computational efficiency, particularly with the adoption of deep learning–based models. Early approaches to object detection were often computationally expensive and time-consuming; however, the introduction of pre-trained convolutional neural networks and optimized architectures has enabled faster and more accurate detection in real-time applications [7–10].

In the context of military and security applications, firearm detection plays a critical role in enhancing situational awareness and reducing risks to both civilian and military personnel. Several studies have explored the use of deep learning techniques for weapon detection, aiming to improve decision-making processes in complex operational environments [11–14]. However, many of these approaches are limited by constrained datasets or lack robustness under varying environmental conditions.

The YOLO (You Only Look Once) family of models has become a dominant approach in real-time object detection due to its ability to achieve a favorable balance between speed and accuracy. YOLOv8, a widely adopted version of this family, introduces architectural improvements that enhance multi-scale feature extraction and detection performance across objects of different sizes [7]. Despite these advancements, challenges remain in detecting small or partially occluded objects in cluttered environments [15–19]. Although newer YOLO variants (e.g., YOLOv9–YOLOv12) have been introduced, YOLOv8 was selected in this study due to its stability, strong community support, and well-documented implementation. Additionally, YOLOv8 provides a favorable balance between detection accuracy and computational efficiency, making it suitable for real-time firearm detection applications.

Several extensions of YOLO-based models have been proposed to address these challenges. For example, the integration of Context Attention Blocks (CAB) has been shown to improve the detection of small objects by leveraging multi-scale feature representations, leading to significant improvements in mean Average Precision (mAP) without increasing computational complexity [8].

In addition, YOLOv8 has been successfully applied in various real-time applications beyond security, including assistive technologies for visually impaired individuals and intelligent traffic monitoring systems. The integration of YOLOv8 with tracking algorithms such as DeepSORT enables robust multi-object detection and tracking in complex environments, even under conditions of occlusion and object overlap [7,8].

Despite these advances, there remains a limited number of studies specifically addressing real-time firearm detection in realistic military scenarios characterized by dynamic conditions, heterogeneous backgrounds, and operational constraints. This gap highlights the need for robust and efficient detection systems tailored to such environments.

The integration of deep learning with advanced sensing technologies presents significant potential for improving firearm detection systems. By leveraging these technologies, it is possible to develop more reliable, accurate, and data-driven solutions that enhance both operational effectiveness and the protection of critical infrastructure [16–20].

III. SYSTEM MODEL AND ALGORITHM

The proposed framework aims to detect small arms and distinguish combatants from non-combatants based on images or video received from compatible streaming devices. Based on the detection results, appropriate actions are triggered. Figure 1 illustrates the overall methodology of the proposed system.

The system consists of several key components: (i) data acquisition module, (ii) small arms dataset, (iii) YOLOv8-based detection module, (iv) decision-making module, (v) feedback and model improvement module, and (vi) system interface. Each component contributes to enabling real-time firearm detection and classification of individuals as combatants or non-combatants.

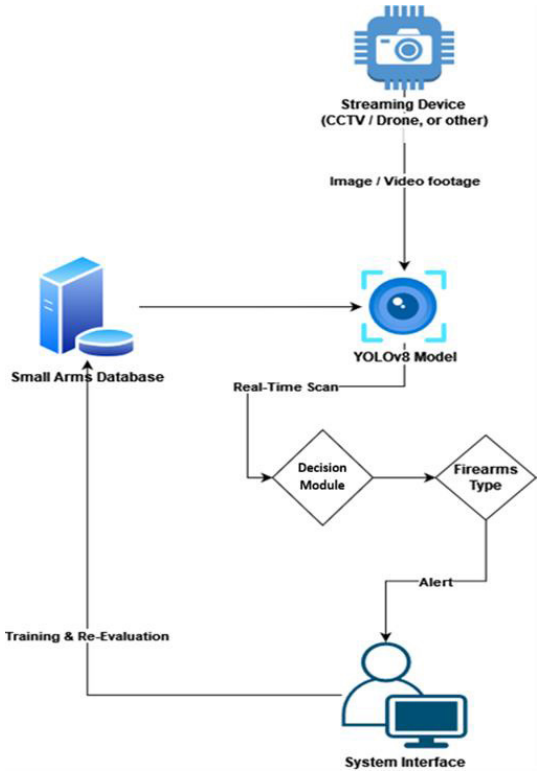


Fig. 1. Diagram overview of the proposed model framework

A. Data Acquisition Module: The system receives input from compatible streaming devices such as CCTV cameras, drones, or other real-time video sources. These devices continuously transmit image or video frames to the processing system for analysis. In this context, a compatible streaming device refers to any device capable of capturing and transmitting real-time video data using standard communication protocols (e.g., RTSP, HTTP, or UDP) and supported video formats.

The device should provide sufficient image resolution (e.g., minimum 720p) and frame rate (e.g., at least 15–30 frames per second) to enable reliable object detection. Examples of compatible devices include CCTV cameras, unmanned aerial vehicles (UAVs), body-worn cameras, and other IP-based video streaming systems.

B. Dataset Construction: A small arms dataset is constructed using verified images of various firearm types, including pistols, rifles, and other small arms.

The images are collected from publicly available and local sources, ensuring diversity in environmental conditions and object appearance.

C. Detection Module (YOLOv8): The YOLOv8 model is trained on the constructed dataset to perform real-time object detection. The model processes incoming frames and identifies the presence of firearms with high accuracy while maintaining real-time performance.

D. Decision-Making Module: Based on the detection results, the system classifies individuals as combatants or non-combatants. If a firearm is detected, predefined actions are triggered, including alert generation and classification outputs.

E. Feedback and Model Improvement: False positives and undetected firearms identified by human operators are collected and fed back into the dataset. This feedback loop enables continuous improvement of the model through retraining and refinement.

F. System Interface: The processed results are displayed through a system interface, providing visual feedback and alerts to the user. This interface supports real-time monitoring and decision-making.

The main contributions of the proposed system are summarized as follows:

1. Design of a real-time firearm detection system based on YOLOv8 for operational environments.
2. Development of a structured dataset for small arms detection under diverse conditions.
3. Implementation of an automated decision-support mechanism for distinguishing combatants from non-combatants.
4. Integration of a feedback mechanism for continuous model improvement through retraining.
5. Deployment-ready architecture compatible with various streaming devices, including CCTV and UAV systems.
6. Applicability of the system in both military and civilian security scenarios.

A. YOLOv8 Model Development and Training Process

To train, validate, and evaluate the proposed model, a customized dataset was constructed. The finalized dataset consists of 3184 annotated images of small arms, including pistols, rifles, and other weapon types. The images were collected from publicly available datasets and Google Images, as summarized in Table I.

TABLE I
DISTRIBUTION OF THE SMALL ARMS DATASET

Dataset Type	Size & Type	Source
Pistol and Knife Dataset	1000 images	Andalusian Research Institute in Data Science and Computational Intelligence
Automatic Weapons Dataset (RPG, Shotgun, SMG, Sniper)	714 images	Snehil Sanyal Weapon Public Dataset
Pistol and Automatic Rifle Dataset	1470 images	A.N.M Jubear Weapons in CCTV Public Dataset

The dataset includes a diverse range of small arms captured in various environments, including day and night conditions, forests, deserts, and urban settings. While the dataset includes diverse weapon types and environmental conditions, the representation of partially occluded objects remains limited. This reflects real-world data availability constraints and highlights an important direction for future dataset expansion.

The images also vary in object characteristics such as size, shape, and color, as illustrated in Figure 2.

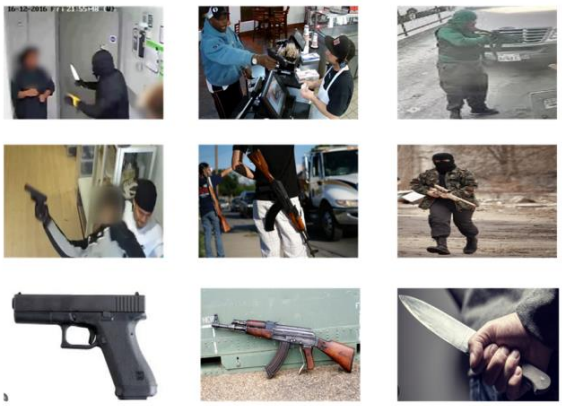


Fig. 2. Example of the images within the complete dataset

In order to successfully adopt and train the dataset in YOLOv8 the dataset itself was assigned labels and the annotating bounding boxes. The value of the annotating box coordinate in each image is then normalized between 0-1 as seen in Figure 3.

Fig. 3. Annotating small arms by class (Knife Class)

This process was carried out using Roboflow, which makes it possible to annotate and make data labels in the desired format. Furthermore, the dataset is carefully divided into three mutually exclusive subsets: training (70%), validation (20%), and test (10%). The training set is used to optimize the model parameters, while the validation set is used for hyperparameter tuning and monitoring the training process. The test set is strictly held out and is not used during training or model selection, ensuring an unbiased evaluation of the model's performance on previously unseen data. Table II presents the class-wise distribution of the dataset, comprising a total of 3184 images.

TABLE II
DATASET ORGANIZED BY CLASSES

Class	Images
All	3184
Pistol	850
Knife	420
Automatic Rifle	700
Shotgun	500
SMG	500
RPG	214

First and foremost, in the model training process we fine-tune the YOLOv8m model which is already pre-trained on the COCO data set. Furthermore, Transfer learning was used to train on the unified dataset. Moreover, various augmentation processes such as HSV, color spacing, mosaic, and image scaling were applied. The model was trained using fine-tuned hyperparameters, including the SGD optimizer, a learning rate of 0.01, and a weight decay of 0.0005, with a batch size of 32. The training process was monitored over multiple epochs, as illustrated in the learning curves presented in Figures 4 and 5.

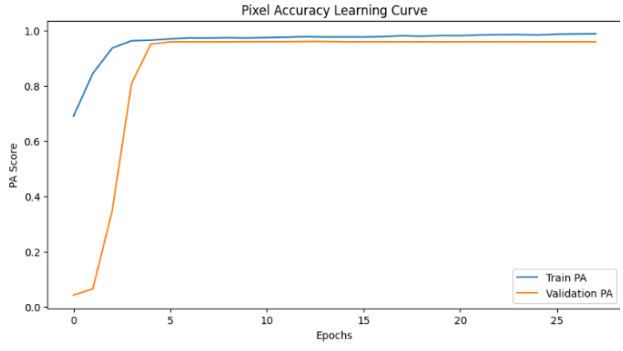
TABLE III
YOLOV8 MODEL TRAINING SUMMARY

Parameter	Value	Description
Model Variant	YOLOv8m	Medium-scale YOLOv8 model
Total Images	2373	Number of training images
Instances	4776	Total labeled objects
Precision (Box(P))	0.938	Bounding box precision
Recall (R)	0.733	Detection recall
mAP@0.5	0.911	Mean Average Precision (IoU = 0.5)
mAP@0.5:0.95	0.569	Mean AP across multiple IoU thresholds

The setup was chosen as it proved very effective in building similar models in previous works related to computer vision target identification [4]. The model was built using Google Collaboratory resources to process the dataset. All performance metrics reported in this study, including precision, recall, and mean Average Precision (mAP), are computed exclusively on the held-out test set. The following results are obtained from evaluation on the held-out test dataset. Care was taken to ensure that no overlapping images or frames were present across the training, validation, and test subsets in order to avoid data leakage. All images were manually verified and annotated to ensure dataset quality and consistency for training and evaluation.

IV. RESULTS AND DISCUSSION

The following results are obtained from evaluation on a strictly held-out test dataset. The training process is illustrated through the learning curves shown in Figures 4 and 5. The accuracy values shown in Figure 4 represent the proportion of correctly classified instances on the validation dataset during training and are presented to illustrate the convergence behavior of the model. The training accuracy increases steadily and stabilizes at a high value, while the validation accuracy follows a similar trend, indicating good generalization capability. Similarly, the training and validation loss decrease and converge, suggesting that the model effectively learns without significant overfitting.



. Fig. 4 Training and Validation Accuracy Curve

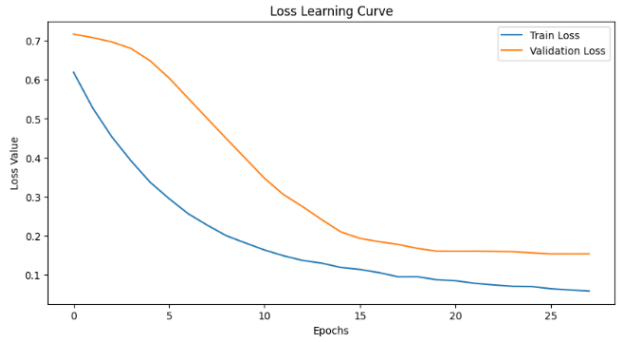


Fig. 5 Learning Loss Curve

To evaluate the performance of the proposed system, the trained model was tested across different scenarios using a confidence threshold of 0.4.

When evaluated on the held-out test dataset, the model achieved an average accuracy of 0.91, reflecting a high proportion of correctly classified instances.

The confusion matrix presented in Figure 6 demonstrates that the model is capable of correctly classifying the majority of samples. Additionally, qualitative results show that the model successfully detects and classifies firearms in various conditions.

TARGET / OUTPUT	Pistol	Knife	Automatic Rifle	Shotgun	SMG	RPG	SUM
Pistol	830 26.07%	15 0.47%	0 0.00%	0 0.00%	5 0.16%	0 0.00%	850 97.65% 2.35%
Knife	10 0.31%	410 12.88%	0 0.00%	0 0.00%	0 0.00%	0 0.00%	420 97.62% 2.38%
Automatic Rifle	0 0.00%	0 0.00%	640 20.10%	10 0.31%	50 1.57%	0 0.00%	700 91.43% 8.57%
Shotgun	0 0.00%	0 0.00%	80 2.51%	400 12.56%	20 0.63%	0 0.00%	500 80.00% 20.00%
SMG	20 0.63%	0 0.00%	50 1.57%	5 0.16%	425 13.35%	0 0.00%	500 85.00% 15.00%
RPG	0 0.00%	0 0.00%	0 0.00%	0 0.00%	0 0.00%	214 6.72%	214 100.00% 0.00%
SUM	860 96.51% 3.49%	425 96.47% 3.53%	770 83.12% 16.88%	415 96.39% 3.61%	500 85.00% 15.00%	214 100.00% 0.00%	2919 / 3184 91.68% 8.32%

Fig. 6. Confusion matrix for classes

Precision, recall, and F1-score were used as primary evaluation metrics. For example, the precision for pistols reaches 0.9765, indicating that approximately 97.65% of detected pistols are correctly classified.

The recall for shotguns is 0.9639, showing that the model successfully identifies most of the actual instances present in the dataset.

The F1-score further confirms balanced performance across classes, with high values observed for pistols (0.9708) and knives (0.9704).

The macro and weighted F1-scores, at 0.9227 and 0.9170 respectively, demonstrate strong overall model performance, with the weighted score accounting for class imbalance in the dataset.

TABLE IV
CLASSIFICATION PERFORMANCE METRICS OF THE PROPOSED MODEL

Class / Metric	Value 1	Value 2
Pistol	Precision: 0.9765	Recall: 0.9651
	F1-score: 0.9708	
Knife	Precision: 0.9762	Recall: 0.9647
	F1-score: 0.9704	
Automatic Rifle	Precision: 0.9143	Recall: 0.8312
	F1-score: 0.8707	
Shotgun	Precision: 0.8000	Recall: 0.9639
	F1-score: 0.8743	
SMG	Precision: 0.8500	Recall: 0.8500
	F1-score: 0.8500	
RPG	Precision: 1.0000	Recall: 1.0000
	F1-score: 1.0000	
Overall Accuracy	0.9168	
Misclassification Rate	0.0832	
Macro F1-score	0.9227	
Weighted F1-score	0.9170	

The obtained results confirm that the proposed YOLOv8-based approach is capable of achieving high detection accuracy across multiple weapon classes, supporting its suitability for real-time deployment in security and surveillance scenarios. These findings support the hypothesis that a YOLOv8-based model can provide reliable detection performance under diverse operational conditions.

However, the model exhibits certain limitations. In some cases, objects such as wallets or other visually similar items may be incorrectly classified as firearms, leading to false positives. These issues are more pronounced at greater distances or under low-quality imaging conditions, where object details are less distinguishable.

Additionally, the detection of small or partially occluded objects remains a challenge, particularly when such cases are underrepresented in the training dataset.

The limited representation of occluded and small objects in the dataset contributes to reduced detection performance in such scenarios, indicating the need for targeted data augmentation and dataset expansion.

Environmental factors such as poor lighting or adverse weather conditions can also impact detection performance.

Fig. 7. Example of predicted output in testing on publicly available CCTV footage

Despite these limitations, the results demonstrate that the proposed system provides a reliable foundation for real-time firearm detection, with potential for further improvement through larger datasets and extended training.

V. EXPERIMENTAL SETUP AND REAL-TIME PERFORMANCE

The proposed system was implemented using a cloud-based environment with GPU acceleration. Specifically, the model was trained and evaluated using an NVIDIA Tesla T4 GPU available through Google Colaboratory. The YOLOv8m model variant was selected due to its balance between detection accuracy and computational efficiency.

The input frames were resized to 640×640 pixels, and standard preprocessing steps were applied, including color space conversion (BGR to RGB) and pixel normalization.
During inference, the system achieved an average processing speed of approximately 20–30 frames per second (FPS) under GPU acceleration, corresponding to a latency of approximately 33–50 ms per frame. This enables near real-time processing of video streams.

The actual performance may vary depending on the computational resources and deployment environment, particularly when processing multiple concurrent video streams.

VI. CONCLUSION

The implementation of a YOLOv8-based system for real-time firearm detection demonstrates promising potential for improving situational awareness and supporting decision-making in dynamic environments such as crowded scenes or operational settings involving potential threats. The model achieves high detection accuracy across multiple weapon classes, indicating its effectiveness in identifying small arms under varying conditions.

The results highlight that YOLOv8 provides a favorable balance between detection accuracy and computational efficiency, enabling near real-time performance when deployed on suitable hardware. This makes the approach applicable to a range of security and surveillance scenarios, including military operations, public safety, and critical infrastructure protection.

However, the experimental results also indicate that detection performance may degrade under challenging conditions, particularly in cases involving partial occlusion, low image quality, or visually complex backgrounds. While the model is capable of detecting many instances under such conditions, its robustness is limited when objects are heavily obscured or insufficiently represented in the training dataset.

These findings suggest that further improvements can be achieved through the use of larger and more diverse datasets, as well as enhancements in model architecture and training strategies. In particular, future work should focus on improving detection performance under occlusion and other adverse conditions through targeted dataset augmentation and specialized model adaptations.

Overall, the proposed approach provides a solid foundation for real-time firearm detection systems, with practical applicability in both military and civilian domains.

ACKNOWLEDGMENT

The author would like to express sincere gratitude to **PhD. Tomislav Shuminoski**, mentor, and **PhD. Aleksandar Risteski**, advisor, for their continuous academic guidance, professional support, and constructive feedback throughout the author's PhD studies. Their expertise, dedication, and encouragement have been instrumental in the author's academic development and in progressing toward the completion of the doctoral research.

REFERENCES

[1] J. C. Scharre, *Army of None: Autonomous Weapons and the Future of War*. New York, NY, USA: W. W. Norton & Company, 2018. DOI: 10.1080/15027570.2019.1637555

[2] M. E. Feickert and A. F. Sadowski, *Artificial Intelligence and National Security*. Congressional Research Service, Washington, DC, USA, Rep., 2021.

[3] Ultralytics, "YOLOv8: A state-of-the-art, real-time object detection model," Ultralytics Documentation, 2023. [Online]. Available: https://docs.ultralytics.com

[4] A. Petrovski, M. Radovanović, and A. Behlić, "Application of drones with artificial intelligence for military purposes," in *Proc. 10th Int. Sci. Conf. Defensive Technologies (OTEH)*, Belgrade, Serbia, 2022, pp. 92–100.

[5] J. Huang *et al*., "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 7310–7311. DOI: 10.48550/arXiv.1611.10012

[6] G. Cheng, P. Zhou, and J. Han, "Learning to predict layout-to-image conditional convolutions for semantic image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 8779–8788. DOI: 10.48550/arXiv.1910.06809

[7] A. Yadav, P. K. Chaturvedi, and S. Rani, "Object detection and tracking using YOLOv8 and DeepSORT," in *Advancements in Communication and Systems*, 2023, pp. 81–90. DOI: 10.56155/978-81-955020-7-3-7

[8] M. Talib, A. H. Y. Al-Noori, and J. Suad, "YOLOv8-CAB: Improved YOLOv8 for real-time object detection," *Karbala Int. J. Mod. Sci.*, vol. 10, no. 1, Art. no. 5, 2024. DOI: 10.33640/2405-609X.3339

[9] C. Zhang, F. Kang, and Y. Wang, "An improved apple object detection method based on lightweight YOLOv4 in complex backgrounds," *Remote Sens.*, vol. 14, Art. no. 3896, 2022. DOI: 10.3390/rs14174150

[10] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 7464–7475. DOI: 10.1109/CVPR52729.2023.00721

[11] M. Maktab, M. Razaak, and P. Remagnino, "Enhanced single-shot small object detector for aerial imagery using super-resolution feature fusion and deconvolution," *Sensors*, vol. 22, no. 12, Art. no. 4562, 2022. DOI: 10.3390/s22124339

[12] L. Deng, H. Li, H. Liu, J. Gu, and L. Deng, "A lightweight YOLOv3 algorithm used for safety helmet detection," *Sci. Rep.*, vol. 12, Art. no. 1125, 2022. DOI: 10.1038/s41598-022-15272-w

[13] L. Ali *et al*., "Development of YOLOv5-based real-time smart monitoring system for increasing lab safety awareness in educational institutions," *Sensors*, vol. 22, no. 15, Art. no. 5555, 2022. DOI: 10.3390/s22228820

[14] H. Liu, F. Sun, J. Gu, and L. Deng, "SF-YOLOv5: A lightweight small object detection algorithm based on improved feature fusion mode," *Sensors*, vol. 22, no. 18, Art. no. 6789, 2022. DOI: 10.3390/s22155817

[15] P. Yong, S. Li, K. Wang, and Y. Zhu, "A real-time detection algorithm based on Nanodet for pavement cracks by incorporating attention mechanism," in *Proc. 8th Int. Conf. Hydraulic and Civil Engineering*, Xi'an, China, 2022, pp. 1245–1250. https://ui.adsabs.harvard.edu/link_gateway/2022ichc.conf...70Y/ DOI: 10.1109/ICHCE57331.2022.10042517

[16] S. Mitra, N. Reddy, I. Tal, and M. Bendechache, "Towards an optimized vehicle detection algorithm for multi-object tracking in traffic surveillance," in *Proc. 29th AIAI Irish Conf. Artif. Intell. Cogn. Sci. (AICS)*, 2021.

[17] Y. R. S. Kumar, T. Nivethetha, P. Priyadharshini, and U. Jayachandiran, "Smart glasses for visually impaired people with facial recognition," in *Proc. Int. Conf. Commun., Comput. Internet Things (IC3IoT)*, Chennai, India, 2022, pp. 1–4.
DOI: 10.1109/IC3IOT53935.2022.9768012

[18] A. Ghosh *et al.*, "Assistive technology for visually impaired using TensorFlow object detection in Raspberry Pi and Coral USB accelerator," in *Proc. IEEE Region 10 Symp. (TENSYMP)*, Dhaka, Bangladesh, 2020, pp. 186–189.
DOI: 10.1109/TENSYMP50017.2020.9230630

[19] P. Boobalan, S. Bhuvanikha, M. Sivapriya, and R. Sivakumar, "Object detection with voice guidance to assist visually impaired using YOLOv7," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 2, pp. 1500–1505, 2023. DOI: 10.22214/ijraset.2023.50182

[20] M. Bianchini, M. Simic, A. Ghosh, and R. N. Shaw, *Machine Learning for Robotics Applications*. Singapore: Springer, 2022.
DOI: 10.1007/978-981-16-0598-7

Rexhep Mustafovski was born in Viterbo, Rome, Italy in 1999. He received the M.Sc. degree in electrical engineering and information technologies from Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia. He is an officer and a dedicated Teaching and Research Assistant in the Department for Cybersecurity and Digital Forensics at the Military Academy "General Mihailo Apostolski" - Skopje. His research interests include military communication systems and platforms, UAV communication systems, cyber resilience, radar systems, and the integration of artificial intelligence and computer vision techniques. He has authored and co-authored over 35 scientific papers published in various international journals and conferences. He presented the scientific paper "MobileSecureComm: A Next-Generation Tactical Communication Platform for Land, Sea and Air Operations" at the 32nd International Conference on Systems, Signals and Image Processing (IWSSIP 2025), held in Skopje in June 2025, with the paper subsequently published in IEEE Xplore.

Aleksandar Risteski received the B.Sc. (1996), M.Sc. (2000), and Ph.D. (2004) degrees in telecommunications from Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia.
He is currently a Professor and Vice- Dean for Research and International Cooperation at the Faculty of Electrical Engineering and Information Technologies at the same university. He has held internships at the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA, where he worked on modeling and optimization of high-speed fiber links. His research interests include optical communications, secure communications, coding theory, and cybersecurity. He has authored and co-authored over 50 journal and conference papers, published widely, including in IEEE Xplore.
Dr. Risteski is a Senior Member of the IEEE and has been an independent member of the Board of Directors of Macedonian Telecom.

Tomislav Shuminoski received the B.Sc. (2008), M.Sc. (2010), and Ph.D. (2016) degrees in electrical engineering and information technologies from the Faculty of Electrical Engineering and Information Technologies, Ss. Cyril and Methodius University in Skopje, Skopje, North Macedonia.
He is currently an Associate Professor at the Institute of Telecommunications within the same faculty. His research focuses on areas such as wireless and mobile multimedia networks, cyber security, and quantum communications and computations. He has published work in international journals and conferences.
Dr. Shuminoski has been a member of the IEEE since 2006.

Optimized Secure Clustering Scheme for Wireless Sensor Networks

Vincent Karovič, Olena Semenova, Maksym Prytula, Volodymyr Martyniuk, and Vladyslav Kuzniak

***Abstract*—Wireless Sensor Networks (WSNs) are increasingly employed in various applications, that require efficient, energy-saving and secure transmitting mechanisms. Clustering is a key technique in WSN to enhance scalability, reduce energy consumption, and manage communication. However, traditional clustering algorithms often lack adaptability to alterable wireless environments. This study presents an optimized secure clustering scheme for Wireless Sensor Networks that employs a fuzzy inference system to perform adaptive and secure cluster head selection. The proposed scheme is designed for WSN with a static topology, where sensor nodes remain fixed after deployment. This assumption matches Internet of Medical Things (IoMT) applications, such as patient monitoring systems and smart hospital infrastructure, where sensors operate in fixed locations. The proposed fuzzy inference system evaluates parameters of sensor nodes to select optimal cluster heads while mitigating malicious node participation. Next, a genetic algorithm was applied to optimize the parameters of the fuzzy inference system, including the membership function shapes and rule base indices, in order to enhance the accuracy of the proposed clustering scheme. MATLAB simulations demonstrate that the proposed scheme effectively identifies compromised sensor nodes and prevents them from assuming the cluster head role. The security validation confirms that the scheme achieves a high detection rate against internal attacks while extending overall network lifetime and maintaining the low latency. This study provides a feasible solution for secure and intelligent clustering in resource- constrained WSN.**

***Index Terms*—Artificial Intelligence, clustering, fuzzy logic, optimization, security, Wireless Sensor Networks**

I. INTRODUCTION

As wireless technologies continue to advance, their range of applications expands correspondingly. Among the various innovative wireless networks, Wireless Sensor Networks (WSNs) have emerged as highly state-of-the-art systems, being implemented in nearly all types of environments, whether rural, suburban, or urban [1]. A WSN can be regarded as an assembly of autonomous spatially distributed sensors that are interconnected and capable of detecting, recording and transmitting variations in environmental physical parameters such as sound levels, temperature, pressure, and motion, while centralizing the collected data. The sensors are compact, intelligent, and small in size, and their data are transmitted wirelessly. WSNs find applications in multiple domains, including healthcare, smart home technologies, industrial automation, and environmental monitoring. Their robustness under extreme conditions renders them a vital element in numerous advanced technological systems, such as automatic irrigation systems, target surveillance, access to clinical records, landslide tracking, and predictive analytics for forest fire and disaster management. In this network topology, two factors – energy and security– are regarded as critical [2]. This is attributed to the features of sensor nodes, which, in contrast to traditional local network devices, are constrained by their battery life and resource limitations, including processing power and bandwidth, as well as frequent deployment in insecure situations [3].

WSNs have been widely applied; however, as the complexity of these networks increases, their security becomes increasingly critical. The primary security vulnerabilities arise from the placement of sensors in isolated locations and their spatial distribution. Unfortunately, conventional security measures are often impractical for sensor networks due to the significant overhead they introduce. Numerous researchers have conducted studies on effective security mechanisms that adhere to the resource and memory limitations inherent in WSNs [4]. Like any communication system, sensor networks require fundamental security services to safeguard data and resources against potential threats. To attain a high level of security, sensor networks necessitate lightweight encryption. Each sensor must strike a balance among cost, performance, and security level; however, achieving this equilibrium is challenging. In certain cases, developers compromise security by opting for cost-effective solutions that lack sufficient mechanisms, such as those for key distribution. A WSN demands more adaptable strategies for key distribution throughout the network. There are two primary approaches: traditional methods utilizing symmetric or asymmetric cryptography, and advanced techniques employing elliptic curve cryptography, which are particularly noted for their

Manuscript received 6 January 2026.

Vincent Karovič is with the Department of Information Management and Business Systems, Comenius University in Bratislava, Bratislava, Slovak Republic (e-mail: vincent.karovic6@fm.uniba.sk).

Olena Semenova and Volodymyr Martyniuk are with the Department of Infocommunication Systems and Technologies, Vinnytsia National Technical University, Vinnytsia, Ukraine (e-mails: semenova.o.o@vntu.edu.ua, vm4ukr@gmail.com).

Maksym Prytula and Vladyslav Kuzniak are with the Department of Information Radioelectronic Technologies and Systems, Vinnytsia National Technical University, Vinnytsia, Ukraine (e-mails: prytula@vntu.edu.ua, kuzniakvl@gmail.com).

DOI: 10.36244/ICJ.2026.2.10

efficient resource utilization compared to other public key methods. Furthermore, WSNs must transmit data securely to a central node through multi-hop communication, ensuring both data confidentiality and integrity. WSNs face various threats and attacks, including data loss, communication disruptions, service degradation leading to delays and denial of service [5, 6]. Consequently, the researchers must focus on primary security attributes of WSN: confidentiality, integrity, authentication.

Therefore, security issues must be carefully considered in multiple processes within WSN, including clustering, routing, and data transmitting. However, although numerous clustering algorithms for WSN have been proposed in the literature, most of them primarily focus on energy efficiency and network lifetime optimization while overlooking security considerations. Moreover, existing clustering algorithms often rely on deterministic methods that lack adaptability to dynamic network conditions and evolving attack patterns. Furthermore, limited attention has been paid to the joint optimization of clustering parameters and intelligent decision-making mechanisms, particularly through the use of novel artificial intelligence techniques. As a result, there remains a research gap in the development of secure and efficient intelligent clustering schemes that simultaneously address energy management, resource balance and resilience against malicious activities. Therefore, there is a need for an intelligent and secure clustering mechanism that simultaneously considers energy consumption and communication behavior of sensor nodes. The problem addressed in this paper is the development of an optimized secure clustering scheme capable of accurately evaluating node eligibility for cluster head selection while incorporating protection against abnormal or malicious behavior. To solve this problem, this paper presents a secure clustering scheme based on Artificial Intelligence techniques for WSNs.

The main contributions of this paper can be summarized as follows: 1) an secure clustering scheme for wireless sensor networks is proposed, integrating energy efficiency and security considerations within an intelligent framework; 2) an intelligent fuzzy inference system is developed to evaluate node eligibility for cluster head selection based on its parameters, enabling security-aware decision making; 3) an optimization algorithm is applied to optimize parameters of the fuzzy inference system, improving clustering accuracy.

Overall, the proposed study improves upon traditional clustering approaches by integrating several AI techniques into the scheme and utilizing security-aware parameters, thereby enhancing both accuracy of cluster head selection and resilience against malicious behavior.

The rest of this paper is organized as follows: we discuss the background in Section II and summarize the related work in Section III. The proposed scheme is described in Section IV and simulated in Section V. Its performance is evaluated in Section VI. Finally, we give the conclusions in Section VII.

II. BACKGROUND

A. Clustering issues in WSN

Clustering methods involve dividing nodes into separate regions known as clusters, each managed by the cluster leader node. By establishing clusters, these networks can achieve required scalability, with the cluster leader node referred to as the cluster head (CH), which coordinates the various regions. The cluster head gathers data from its member nodes, aggregates this information, and transmits it to the base station (BS). Data transmission from CHs to the BS can be performed with single-hop or multi-hop techniques. A CH may be either designated by the sensors within the clusters or predetermined by network designers. Various clustering algorithms have been developed to promote effective communication in WSNs. Notable clustering algorithms include LEACH [7], PEGASIS [8], TEEN [9], and APTEEN [10]. In the context of WSN, clustering has shown notable energy efficiency by enhancing network lifetime, bandwidth, and reducing energy consumption.

Clustering algorithms can be classified into two primary categories: distributed and centralized methods. Distributed systems use local information to form clusters, while centralized methods utilize the global knowledge of the network. The pioneering method for cluster head selection is the Low Energy Adaptive Clustering Hierarchy (LEACH) protocol, which is based on a probabilistic approach [11]. It is noteworthy that each node in the system generates a random number, and if this number is below a specified threshold, the corresponding node will identify itself as the cluster head and send an advertisement message to all other sensor nodes in the network. Each non-cluster head node that receives this announcement considers the communication impact before deciding which cluster head to connect with.

Energy-efficient direction-finding approaches are of significance in the context of reducing energy expenditure, enhancing network efficiency, and prolonging the operational lifespan of networks. This is due to the fact that sensor nodes and base stations engage in the transmission of packets via wireless radio signals [12]. In the clustering process, a direct communication between each node and the CH is to be established. Subsequently, the CH disseminates the aggregated, compressed, and transmitted data to the BS or an alternative CH in the vicinity [13]. It has been demonstrated that a greater quantity of energy is expended when transmitting the same data to the BS directly than when sending it in stages over smaller distances. Consequently, researchers have directed their attention towards the field of clustering.

B. Artificial Intelligence

Traditional clustering algorithms often struggle with dynamic environments and uncertain data. Artificial Intelligence (AI) provides heuristic approaches that may overcome the limitations of traditional deterministic or probabilistic methods [14]. AI techniques, namely Fuzzy Logic, Neural Networks, and Genetic Algorithms, may offer adaptive and intelligent solutions for clustering problem in WSNs. They

have been successfully employed to improve routing, load balancing, energy efficiency and security [15].

Fuzzy logic mimics human reasoning when dealing with uncertain or imprecise data using linguistic variables and fuzzy sets [16]. In WSN clustering, parameters like residual energy, node centrality, and distance to the base station often have vague boundaries, that is why it may benefit from fuzzy representation. Thus, in fuzzy-based clustering, a fuzzy inference system (FIS) can be applied to evaluate node suitability as a CH. The FIS processes all input variables using fuzzy rules to calculate a clustering score or CH suitability index for each node. Furthermore, security-aware fuzzy rules can ensure that only trusted nodes with sufficient resources are elected.

Neural networks (NNs) are computational models inspired by biological neurons and capable of learning patterns and relationships from real data. Traditional clustering methods often rely on predefined rules and algorithms. However, these methods may not adapt well enough to changeable network conditions or complex security threats. NNs, with their ability to learn from data, offer a promising solution for effective CH selection. In WSNs, NNs can learn optimal strategies for CH selection and clustering considering parameters of the nodes and the network. NNs are trained using these data to predict the best nodes for CH roles and optimal cluster sizes. Security-aware training includes identifying and penalizing nodes that exhibit malicious behavior. Neural networks can enhance WSN security by analyzing statistical data to assign trust scores to nodes. Thus, anomalous behavior can reduce a node's trust and CH candidacy.

Genetic Algorithms (GAs) are evolutionary optimization techniques inspired by natural selection processes. In clustering problems, chromosomes of genetic algorithm represent possible cluster configurations or CH assignments. The GA iteratively evolves better clustering schemes reaching an optimal or near-optimal configuration. When applying a GA, sensor nodes with suspicious behavior or low trust scores are penalized in the fitness function, reducing their chance of becoming CHs.

III. RELATED WORKS

Various countermeasures have been suggested by researchers to ensure efficient and energy-saving clustering in WSN. Secure and reliable clustering schemes for WSN are considered and discussed in [17]. Study [18] introduces an effective fuzzy-based method for continuous node refinement, leveraging the multilayer routing architecture of WSNs. This work aims to enhance the reliable data transmission in cross-layer WSN. Unsuitable sensor nodes are identified based on their fuzziness. However, this study considers only residual energy of a node, neglecting traffic parameters of the network. An Energy Adaptive Distributed Energy-Efficient Clustering approach was proposed in [19] to be employed to ensure the resilience of WSNs throughout their operational lifespan while also improving the efficacy of their device nodes. Here, the main objective was lifetime of the sensor nodes in WSNs, and the further recommendation was for longevity while security issues have not been considered.

Paper [20] employs an ensemble methodology to create an improved hierarchical procedure for low-energy clustering, with the main objective of refining and optimizing the LEACH approach. The comparison results show that the improved method lengthens the network's lifetime by balancing the energy in each cluster. While the traditional LEACH method has been improved, no novel technology has been proposed.

Study [21] proposes a distributed protocol that establishes a cluster architecture and selects a rendezvous node with sufficient power for prolonged operation, situated in proximity to the mobile sink. A novel weight-based tree-construction method is introduced. The clustering approach of this algorithm creates a framework comprising clusters of varying sizes. Nevertheless, the proposed approach in suitable only for application in the Industrial WSN.

Work [22] presents an energy aware fuzzy-neuro clustering algorithm utilizing three key metrics for selecting cluster heads: residual energy, distance and node degrees. The algorithm comprises two main stages, namely clustering and multi-hop routing. Although the MATLAB simulation demonstrate the superiority of the proposed algorithm over the existing ones, it considers no communication security issues that restricts its utilization in complex environment of IoT. Paper [23] develops a clustering algorithm for networks that consumes less power. In this algorithm, few settings are be set before the cluster can be restarted. The clustering cycles operate under the assumption that all nodes possess an equal amount of energy. This algorithm identifies the node designated as the cluster head by evaluating the remaining energy, distance to the base station, and proximity to other members of the cluster. But its serious drawback is a greater amount of control messages. Study [24] suggests a fuzzy logic-based energy efficient clustering hierarchy where clustering is performed within a non-uniform WSN environment, with each node calculating a probability value based on three fuzzy input parameters, namely a node's centrality, residual energy and distance from the base station. The suggested protocol has two main stages – the formation of clusters and the collection of data. The drawback is that the nodes mainly join the closest to them cluster head, i.e., the key metric is the distance.

Paper [25] discusses a hybrid approach designed for the energy-efficient clustering of objects varying in size. Here, the clustering is to be performed across numerous layers. The layers are developed in a base station, which also applies neighboring node information. For data transmission both inter-cluster and intra-cluster directing is used. A significant improvement is a reduction in control messages. Simulation results prove that the proposed method has increased network stability and lifetime of the network. However, such hybrid clustering is quite complicated in implementation, that restricts its area of application.

Fuzzy algorithms for the election of cluster heads in cluster-based hierarchical routing protocols were developed in [13]. Instead of relying on a comprehensive aggregation of factors like remaining energy, proximity to the base station, resident detachment, concentration, and significance, each of these

fuzzy sensing-based frameworks utilizes a selective combination of these attributes in the process of choosing cluster heads. However, due to large number of attributes these algorithms require enhanced memory resources, which makes its implementation challenging. Work [26] proposes to utilize the fuzzy c-means clustering algorithm to establish k optimal clusters. The responsibility for determining the optimal number of clusters is assigned to the base station, which possesses greater computational capabilities and a global view of the network. A key strength of the proposed algorithm is that the initial clusters are predefined, and the subsequent procedure focuses on selecting cluster heads only from these verified and secure groups. This design provides an advantage over many conventional approaches, where cluster formation and CH selection are performed simultaneously. In the CH selection process for each cluster, residual energy and communication cost are used as fuzzy input variables. The drawback of this algorithm is a high computational overhead.

Paper [27] introduces a protocol that builds upon the LEACH protocol by incorporating both energy levels and the distances of nodes within WSN for the selection of CHs. Several methods were suggested to obtain the throughput of contention-free and contention-based protocols. A comparison between different protocols of the same class was performed. A new method to increase the efficiency of LEACH protocol was proposed. Nonetheless, this protocol is applicable solely in scenarios where a BS is present within the sensor field.

The conducted literature review demonstrates that a substantial number of studies have been devoted to solving the clustering problem in WSNs, whereas particular attention is given to modern approaches, especially such as artificial intelligence technologies as recent advances in intelligent methods provide new opportunities to enhance clustering efficiency beyond traditional techniques. However, when addressing the clustering problem, it is essential to simultaneously consider not only energy communication, but also communication traffic features and security issues in WSNs, as they have a significant impact on the reliability of the overall system [28].

While the methodologies presented in [13], [22], and [24] also utilize fuzzy inference systems to compute the chance of sensor nodes for CH election, the optimized secure clustering scheme proposed in this study distinguishes itself by addressing energy efficiency and network security. Unlike the aforementioned approaches, which depend on energy consumption and node's spatial positioning, our scheme integrates traffic-centric parameters, namely, the transmitting and delay ratios. The inclusion of these behavioral indicators promotes the indirect detection of anomalous activity, thereby reducing the risk of assigning the CH role to compromised or malicious nodes.

IV. PROPOSED METHODOLOGY

A. Secure clustering scheme

We propose an optimized secure clustering scheme for WSNs, as illustrated in Fig. 1. The overall process consists of three stages: data processing, intelligent cluster head selection, and cluster formation.

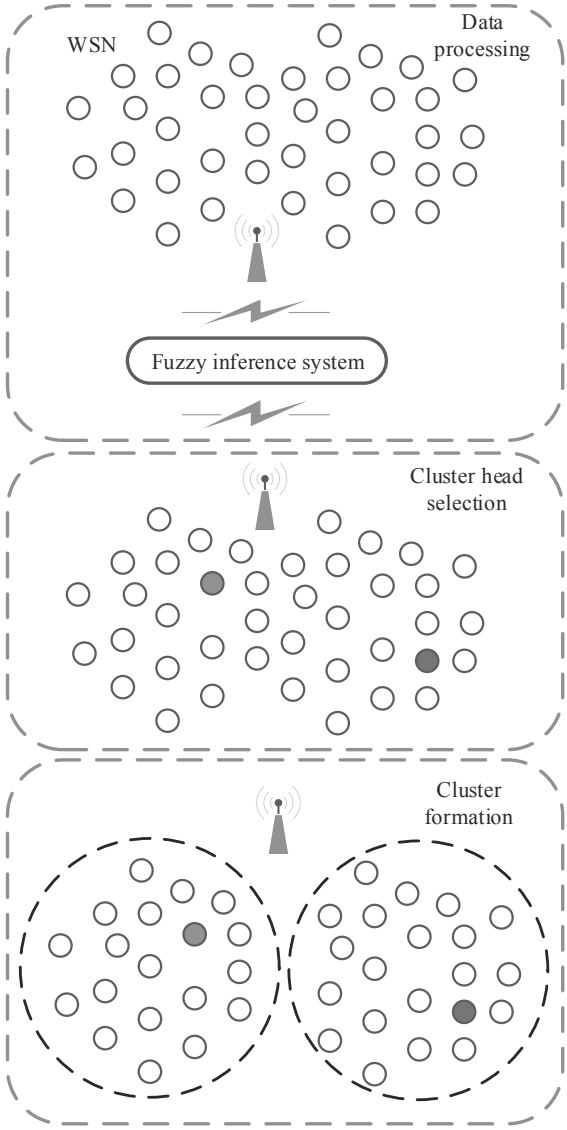


Fig. 1. Optimized secure clustering scheme

Initially, sensor nodes are deployed within the monitoring area and perform data acquisition. Each sensor node must be able to communicate directly with a BS. In IoMT wireless body area networks, this assumption is quite realistic because the distance between body sensors and the local gateway, such as a smartphone or smart hub, is usually only a few meters. During the data processing phase, key parameters describing each node's state are collected and forwarded to the BS for further evaluation. The BS runs the FIS, which serves as the decision-making component of the proposed scheme. By processing the input parameters, it evaluates the suitability (chance index) of each node to act as a cluster head. This index represents the degree to which a node is suitable for assuming the role of cluster head, taking into account not only energy efficiency but also traffic behavior and potential security anomalies. Based on

the computed chance indexes, nodes with the highest values are selected as cluster heads in the cluster head selection phase, as shown in the middle part of the figure. Once the cluster heads are determined, the network proceeds to the cluster formation stage. Ordinary sensor nodes associate with the nearest trusted cluster head in terms of distance, forming logical clusters represented by dashed boundaries in the figure. Within each cluster, member nodes transmit their data to the corresponding cluster head. The cluster heads perform data aggregation to reduce redundancy and securely forward the aggregated information to the base station, thereby enhancing both efficiency and network security.

B. Optimized fuzzy inference system

The core of the proposed optimized secure clustering scheme is a fuzzy inference system.

Fig. 2 illustrates the structure of the FIS used for secure cluster head selection. The left-hand side of the diagram presents the standard processing flow of a Mamdani-type FIS [29]. The process begins with the block labeled "Data for selecting Cluster Head," which represents the crisp input parameters. These inputs are first processed in the fuzzification unit, where they are converted into degrees of memberships by using linguistic terms and membership functions and then forwarded to the inference engine. The inference engine operates based on the rule base, which contains the set of fuzzy if–then rules provided by the experts. Inference engine performs a fuzzy reasoning to map inputs to outputs considering these rules. The aggregated fuzzy output is passed to the defuzzification unit, where it is transformed into a crisp numerical value labeled as the "Chance index." Fig. 2 also illustrates the optimization mechanism. The genetic algorithm, shown on the right side of the diagram, is connected to the rule base, indicating that it is responsible for optimizing the parameters of the FIS, that involves fine-tuning membership functions and adjusting parameters of fuzzy rules to enhance system's performance. As medical sensor nodes have limited battery power, memory, and processing capabilities, the FIS is executed by the BS, which has greater processing power and energy resources. In a real IoMT environment, the BS can be a smartphone, a wearable smart hub, a bedside monitor, or a local hospital gateway.

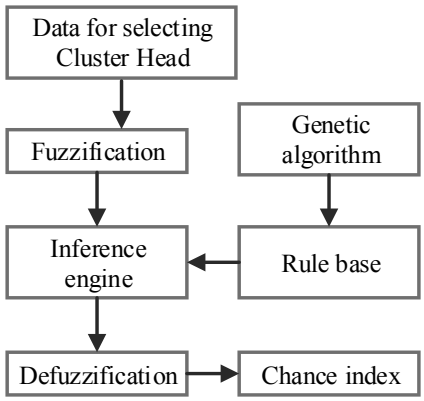


Fig. 2. Optimized fuzzy inference system for secure clustering

C. Parameters of the optimized FIS

The output value of the FIS is the chance index of a sensor node (Ch). It refers to the computed probability of the node to be selected for a cluster head, based on input criteria, that determine its priority.

The proposed FIS has three inputs: residual energy (RE), transmitting ratio (TR), and delay ratio (DR).

The selection of input parameters is motivated by their strong relevance to both network performance and security issues. Although the proposed scheme does not explicitly include a dedicated trust score as a direct input to the FIS, it inherently captures security-related behavior through its input parameters. These parameters act as indirect yet effective indicators of node trustworthiness and reliability, enable the scheme to infer suspicious activity through deviations in communication and energy patterns. Together, these parameters provide complementary indicators, enabling the FIS to detect suspicious behavior of a node. By embedding these behavior-based cues into the fuzzy inference process, the scheme evaluates node credibility, avoiding the overhead associated with maintaining reputation management mechanisms, while still enhancing clustering security.

The residual energy of a sensor node is the amount of remaining battery power available in it. Anomalous depletion of residual energy in a sensor node may indicate the presence of denial-of-service (DoS) or flooding attacks, as such behavior often results from malicious activities that force the node to perform excessive communication or computation.

The transmitting ratio of a sensor node is the proportion of data packets successfully transmitted by the node compared to the total number of packets it was supposed to send. An abnormally high transmitting ratio in a sensor node indicate a network attack, as it suggests the node is being exploited to send excessive data, potentially due to malicious control or flooding-based attacks. Conversely, an abnormally low TR may be related to gray hole attack, black hole attack or replay attack.

The delay ratio of a sensor node is the ratio of the actual end-to-end delay experienced by data packets to the expected or predefined optimal delay. An elevated delay ratio in a sensor indicates a network attack such as selective forwarding or a replay attack, as it reflects increased latency potentially caused by congestion, malicious interference, or deliberate disruption of communication paths.

We have chosen for the inputs such linguistic terms as: "low", "medium", and "high" for RE; "small", "moderate", and "large" for TR and DR. This can be regarded as intuitive categories associated with general assessment terms and facilitates decision-making.

The input values are determined by membership functions, which are of a trapezoid type, which is computationally simple and provides smooth transitions between fuzzy sets, making it well suited for such resource-constrained systems as WSN. Moreover, in this study, normalized input values are used to map all variables into a common range that ensures consistency in fuzzy inference and prevents dominance of any single input.

The residual energy input value is defined as:

$$\mu_{low}(RE)=\begin{cases}1 \;\; if \;\; RE\le 10,\\ \dfrac{30-RE}{30-10} \;\; if \;\; 10<RE\le 30,\\ 0 \;\; if \;\; RE>30.\end{cases} \quad (1)$$

$$\mu_{med}(RE)=\begin{cases}0 \;\; if \;\; RE\le 10,\\ \dfrac{RE-10}{10-30} \;\; if \;\; 10<RE\le 30,\\ 1 \;\; if \;\; 30<RE\le 50,\\ \dfrac{70-RE}{70-50} \;\; if \;\; 50<RE<70,\\ 0 \;\; if \;\; RE\ge 70.\end{cases} \quad (2)$$

$$\mu_{high}(RE)=\begin{cases}0 \;\; if \;\; RE\le 50,\\ \dfrac{RE-50}{70-50} \;\; if \;\; 50<RE\le 70,\\ 1 \;\; if \;\; RE>70.\end{cases} \quad (3)$$

The residual energy represents the remaining battery energy of a sensor node and is expressed as a percentage of the initial node energy, ranging from 0% to 100%. The threshold values used in (1)–(3) were selected to divide the energy range into three operational regions corresponding to low, medium, and high energy levels. Nodes with RE below 30% are considered energy-constrained and therefore less suitable for the cluster head role due to the increased risk of premature energy depletion. The medium-energy region 30–70% represents nodes with sufficient energy to participate in network operations, while nodes with RE above 70% are considered highly suitable candidates for cluster head selection. These thresholds were chosen based on commonly adopted energy management principles in WSN clustering, where cluster heads are preferably selected from nodes with relatively high residual energy to balance energy consumption and extend network lifetime.

The transmitting ratio input value is defined as:

$$\mu_{sm}(TR)=\begin{cases}1 \;\; if \;\; TR\le 20,\\ \dfrac{40-TR}{40-20} \;\; if \;\; 20<TR\le 40,\\ 0 \;\; if \;\; TR>40.\end{cases} \quad (4)$$

$$\mu_{mod}(TR)=\begin{cases}0 \;\; if \;\; TR\le 20,\\ \dfrac{TR-20}{40-20} \;\; if \;\; 20<TR\le 40,\\ 1 \;\; if \;\; 40<TR\le 60,\\ \dfrac{80-TR}{80-60} \;\; if \;\; 60<TR<80,\\ 0 \;\; if \;\; TR\ge 80.\end{cases} \quad (5)$$

$$\mu_{lar}(TR)=\begin{cases}0 \;\; if \;\; TR\le 60,\\ \dfrac{TR-60}{80-60} \;\; if \;\; 60<TR\le 80,\\ 1 \;\; if \;\; TR>80.\end{cases} \quad (6)$$

The delay ratio input value is defined as:

$$\mu_{sm}(DR)=\begin{cases}1 \;\; if \;\; DR\le 30,\\ \dfrac{50-DR}{50-30} \;\; if \;\; 30<DR\le 50,\\ 0 \;\; if \;\; DR>50.\end{cases} \quad (7)$$

$$\mu_{mod}(DR)=\begin{cases}0 \;\; if \;\; DR\le 30,\\ \dfrac{DR-30}{50-30} \;\; if \;\; 30<DR\le 50,\\ 1 \;\; if \;\; 50<DR\le 70,\\ \dfrac{90-DR}{90-70} \;\; if \;\; 70<DR<90,\\ 0 \;\; if \;\; DR\ge 90.\end{cases} \quad (8)$$

$$\mu_{lar}(DR)=\begin{cases}0 \;\; if \;\; DR\le 70,\\ \dfrac{DR-70}{90-70} \;\; if \;\; 70<DR\le 90,\\ 1 \;\; if \;\; DR>90.\end{cases} \quad (9)$$

For the output, such linguistic terms as “very weak”, “weak”, “average”, “strong”, “very strong” were chosen.

The output value of chance index is determined by membership functions, which are of triangular type, as:

$$\mu_{vw}(Ch)=\begin{cases}1 \;\; if \;\; Ch\le 0,\\ \dfrac{25-Ch}{25} \;\; if \;\; 0<Ch\le 25,\\ 0 \;\; if \;\; Ch>25.\end{cases} \quad (10)$$

$$\mu_{w}(Ch)=\begin{cases}0 \;\; if \;\; Ch\le 0,\\ \dfrac{Ch}{25} \;\; if \;\; 0<Ch\le 25,\\ \dfrac{50-Ch}{50-25} \;\; if \;\; 25<Ch\le 50,\\ 0 \;\; if \;\; 50<Ch.\end{cases} \quad (11)$$

$$\mu_{av}(Ch)=\begin{cases}0 \;\; if \;\; Ch\le 25,\\ \dfrac{Ch-25}{50-25} \;\; if \;\; 25<Ch\le 50,\\ \dfrac{75-Ch}{75-50} \;\; if \;\; 50<Ch\le 75,\\ 0 \;\; if \;\; 75<Ch.\end{cases} \quad (12)$$

$$\mu_{st}(Ch)=\begin{cases}0 \;\; if \;\; Ch\le 50,\\ \dfrac{Ch-50}{75-50} \;\; if \;\; 50<Ch\le 75,\\ \dfrac{100-Ch}{100-75} \;\; if \;\; 75<Ch\le 100,\\ 0 \;\; if \;\; 100<Ch.\end{cases} \quad (13)$$

$$\mu_{vst}(Ch)=\begin{cases}0 \;\; if \;\; Ch\le 75,\\ \dfrac{Ch-75}{100-75} \;\; if \;\; 75<Ch\le 100,\\ 1 \;\; if \;\; Ch>100.\end{cases} \quad (14)$$

The proposed secure clustering FIS operates according to 27 fuzzy rules, which define the relationship between input data and the corresponding control action (Table 1). The fuzzy rule

base was constructed using an analytical evaluation of sensor node characteristics under various network conditions. The linguistic rules were formulated to reflect intuitive relationships between node's parameters, and its suitability for acting as a cluster head. The rule weights were initially set uniformly to ensure balanced influence of all rules and will be subsequently refined during the optimization process.

TABLE I
FUZZY RULES

Rule	RE	TR	DR	Ch
1	Low	Small	Small	Weak
2	Low	Small	Moderate	Very weak
3	Low	Small	Large	Very weak
4	Low	Moderate	Small	Very weak
5	Low	Moderate	Moderate	Very weak
6	Low	Moderate	Large	Very weak
7	Low	Large	Small	Weak
8	Low	Large	Moderate	Weak
9	Low	Large	Large	Weak
10	Medium	Small	Small	Average
11	Medium	Small	Moderate	Weak
12	Medium	Small	Large	Weak
13	Medium	Moderate	Small	Average
14	Medium	Moderate	Moderate	Average
15	Medium	Moderate	Large	Average
16	Medium	Large	Small	Strong
17	Medium	Large	Moderate	Strong
18	Medium	Large	Large	Average
19	High	Small	Small	Strong
20	High	Small	Moderate	Strong
21	High	Small	Large	Strong
22	High	Moderate	Small	Very strong
23	High	Moderate	Moderate	Very strong
24	High	Moderate	Large	Very strong
25	High	Large	Small	Very strong
26	High	Large	Moderate	Very strong
27	High	Large	Large	Strong

V. COMPUTER SIMULATION

A. Simulation of FIS

To validate the operability of the developed secure clustering FIS, we have utilized the Fuzzy Inference System Toolbox in MATLAB program, which has powerful Graphical User Interface. We assigned the inputs and output, and specified the rule base.

To check the functionality of the FIS, a series of simulations was carried out to generate the required output values, during which multiple experiments with various values of input parameters were performed.

Fig. 3 illustrates the outcome of one simulation of the FIS obtained in MATLAB for a specific set of input parameters. Each row corresponds to a fuzzy rule from the rule base provided in Table 1. The first three columns show the membership functions associated with the input variables (residual energy, transmitting ratio, and delay ratio) according to (1)–(9), while the last column shows the membership function of the output variable (the chance index) according to (9)–(14). The vertical lines indicate the current crisp input values, and the dashed horizontal lines represent the corresponding membership degrees. The shaded regions in the output membership functions represent the activation of the fuzzy rules. The final defuzzified value is displayed at the right top of the figure. A detailed description of the rule viewer and fuzzy inference process can be found in the MATLAB Fuzzy Logic Toolbox documentation [30].

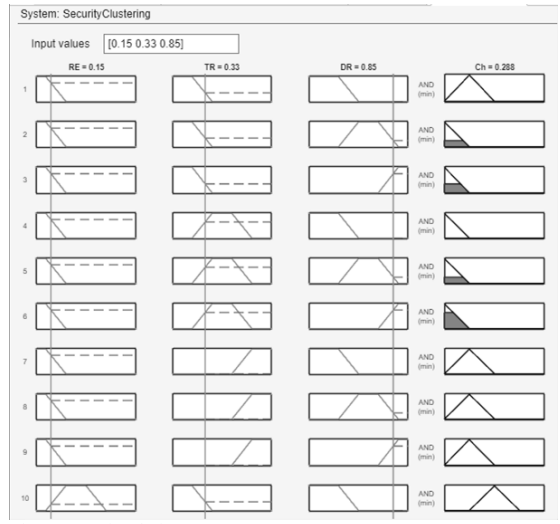


Fig. 3. FIS simulation

In this simulation, the residual energy equals 0.15, the transmitting ratio equals 0.33, the delay ratio equals 0.85. The calculated by the FIS chance index equals 0.288, which corresponds to a low suitability level for cluster head selection.

After having regarded the results of the simulations, the authors concluded that the simulated FIS operates properly.

B. FIS Optimization

In this investigation, we optimized the parameters of the developed fuzzy inference system using a genetic algorithm in MATLAB. The optimization process aims to enhance the FIS performance by tuning membership functions and rule weights. MATLAB software provides a suitable environment for integrating fuzzy logic systems with optimization techniques. This approach is expected to yield improved control accuracy and system stability.

GAs were chosen as optimization technique due to their capability to handle nonlinear search spaces. The MATLAB GA tool iteratively generates populations of parameter sets (called chromosomes), applies selection, crossover, and mutation operations, and evolves toward an optimal solution. The process continues until convergence criterion is met, after which the best-performing parameter set is selected for the final implementation of the fuzzy inference system.

The chromosome for the developed FIS combines the parameters of all membership functions with the numerical weights of the fuzzy rule base. The chromosome can be expressed as a vector $\mathbf{p} = [\mathbf{q} \mid \mathbf{r}]$, where the first segment $\mathbf{q}$ contains the parameters of all input and output membership functions, whereas the second segment $\mathbf{r}$ encodes the structure of 27 fuzzy rules.

The developed FIS contains three input variables each with

three trapezoid membership functions encoded as

$$q_{in,m}=[q_{4(m-1)+1}, q_{4(m-1)+2}, q_{4(m-1)+3}, q_{4(m-1)+4}] \quad (15)$$

and one output variable with five membership functions encoded as

$$q_{out,t}=[q_{36+3(t-1)+1}, q_{36+3(t-1)+2}, q_{36+3(t-1)+3}]. \quad (16)$$

Altogether, the membership-function segment contains 51 parameters (genes):

$$q=[q_{in,1},\ldots,q_{in,9},q_{out,1},\ldots,q_{out,5}]. \quad (17)$$

The remaining part of the chromosome encodes the fuzzy rule base using 108 numerical genes. Each of the 27 rules is represented by four genes: three genes corresponding to the three input variables and one gene describing the output variable. Thus, rule k is encoded as

$$r_k=(r_{4(k-1)+1}, r_{4(k-1)+2}, r_{4(k-1)+3}, r_{4(k-1)+4}). \quad (18)$$

The full chromosome is the concatenation of the MF-parameter block and the rule-base block:

$$\mathbf{p}=[q_{in,1},\ldots,q_{in,9},q_{out,1},\ldots,q_{out,5},r_1,...,r_{27}]. \quad (19)$$

For the optimization process, a training dataset $\{(x_i, y_i)\}_{i=1}^{M}$ was assigned. For each sample x_i the fuzzy system produces an output $\hat{y}_i = f_{\mathrm{FIS}}(x_i; p)$ obtained through fuzzification, rule aggregation and defuzzification with chromosome p.

The quality of a chromosome is quantified through the objective function that measures the discrepancy between the FIS output with parameters p, denoted as $f_{\mathrm{FIS}}(x_i; p)$ and the desired output y_i. In this study the root mean squared error (RMSE) is used as the fitness measure. So, the objective of the optimization is to minimize the RMSE:

$$J(p)=\sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(y_i - f_{\mathrm{FIS}}(x_i; p)\right)^2}. \quad (20)$$

Thus, the optimization problem is:

$$p^* = \underset{p_{\min}\le p\le p_{\max}}{\operatorname{argmin}} \; J(p). \quad (21)$$

The final optimized FIS is obtained by substituting the optimal chromosome:

$$\mathrm{FIS}^* = f_{\mathrm{FIS}}(p^*). \quad (22)$$

This optimized system represents the best configuration of membership functions and rule base parameters found by the genetic algorithm with respect to the training dataset. This optimized system minimizes the prediction error on the training dataset and provides an improved model for estimating the priority of sensor nodes.

For the GA optimization process, the population size was 159 individuals. A crossover rate of 0.8 was applied to promote effective information exchange between parent solutions, while a mutation rate of 0.05 was used to preserve genetic diversity and avoid premature convergence. The training dataset used for optimization was generated through simulation by varying the input parameters within their normalized ranges and computing the corresponding desired output values based on predefined performance criteria. The original RMSE value was 216×10^{-6} and after the optimization the RMSE value was 176.6×10^{-6}

Fig. 4 presents the training convergence curve of the genetic algorithm applied to optimize the parameters of the FIS within the proposed secure clustering scheme in MATLAB. The horizontal axis represents the generation number, the vertical axis shows the value of the optimization cost (minimum fitness value). The fitness function corresponds to the objective minimized during training, reflecting the error between the desired and obtained FIS outputs. At the initial generations, the optimization cost is relatively high, indicating that the randomly initialized population does not yet provide an optimal parameter configuration. The overall decreasing trend of the curve confirms the stability and effectiveness of the genetic optimization process in refining the membership functions and rule parameters of the FIS. As shown in the figure, the genetic algorithm terminated at generation 52. This confirms that the algorithm has reached a near-optimal solution.

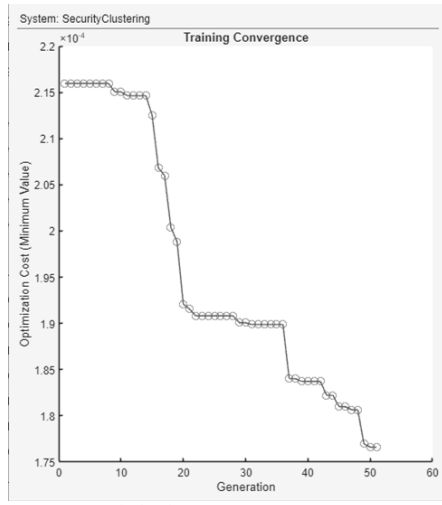


Fig. 4. FIS optimization

C. FIS validation

The developed secure clustering FIS was validated in MATLAB environment by applying the System Validation Tool. The validation was performed on a dataset generated considering the data from the WSN-DS dataset. Fig. 5 presents the validation results of the developed FIS, that is its prediction accuracy. The upper plot illustrates the variation of three input variables across the validation dataset. Here, the inputs demonstrate diverse amplitudes, confirming that the validation was performed on the dataset containing sufficiently varied operating conditions. The horizontal axis, denoted as the Data Point Index, represents the sequential number of a validation sample. The Output corresponds to the chance index value. The bottom plot compares the reference output values from the dataset with the outputs predicted by the FIS. Considering the close alignment between two curves on the bottom plot, we may conclude that there is a strict correspondence between the expected and actual output responses, thus indicating reliable FIS performance under test conditions. Overall, the plot illustrates the ability of the FIS to estimate the cluster-head eligibility index for a node that is subsequently used during the clustering process.

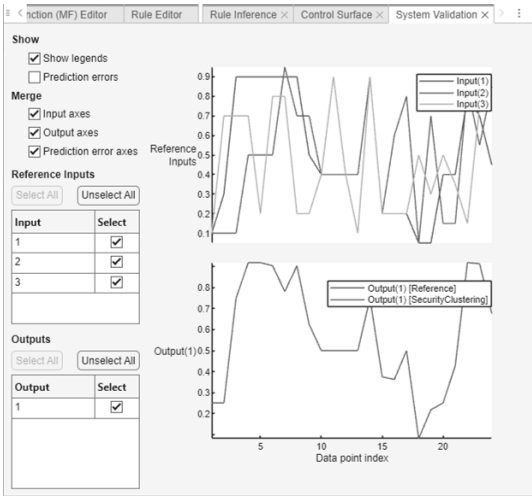


Fig. 5. FIS validation

VI. EVALUATION

The proposed optimized secure clustering scheme can be employed in the Internet of Medical Things, to organize medical sensor nodes into secure clusters, ensuring reliable monitoring of patient health data. By incorporating security metrics in the decision-making process, the scheme safeguards sensitive medical information from unauthorized access and potential cyber threats, that is essential for IoMT [31].

Although some IoMT applications may involve mobile devices, the proposed framework assumes a relatively static wireless body area network. Two options are possible. The first is patient monitoring in intensive care units, where patients remain in fixed locations and both the sensors and the local gateway are stationary. The second is wearable health monitoring, where a patient may move, but the relative positions of the body sensors and the local cluster head (such as a smartphone or wearable hub) remain nearly unchanged. As a result, communication distances between nodes are stable, allowing the proposed fuzzy clustering scheme to improve energy efficiency and security without the need for frequent route updates.

To evaluate the operability of the proposed clustering scheme, a simulation was conducted in the MATLAB environment. The simulated model assumes a IoMT wireless sensor network, which may be applied for continuous patient monitoring. The network is composed of 100 medical sensor nodes transmitting sensitive physiological data and randomly deployed over a monitoring area of 100×100m. The CHs aggregate the data and forward it to an external BS. Initial energy of each sensor node equals 0.5 J. Packet size of transmitted data is 4000 bits.

The adversary assumptions consider compromised internal nodes capable of launching attacks such as blackhole, flooding, replay attack. For the computer simulation the attack scenario was as follows. Node #10 was modeled as executing a blackhole attack: it participates in routing, receives many packets from neighbors, but intentionally drops them and transmits almost nothing further. This results in a very low transmitting ratio of the node. Node #25 was modeled as executing a flooding attack by generating a huge number of transmission requests, that leads to increase in its transmitting ratio and accelerated energy depletion of neighboring sensor nodes. Node #40 was modeled as executing a replay attack, where previously captured packets were retransmitted to create artificial traffic patterns, this increases delay ratios.

The operability of the proposed fuzzy clustering scheme was compared with that of the conventional LEACH protocol under the discussed attack scenario. Fig.6 and Fig. 7 present the MATLAB simulation results of cluster head selection in LEACH protocol and in the proposed fuzzy-based secure clustering scheme respectively. Here, circles represent sensor nodes randomly deployed in the sensing field, crosses represent nodes under attack, stars represent nodes assigns as cluster heads.

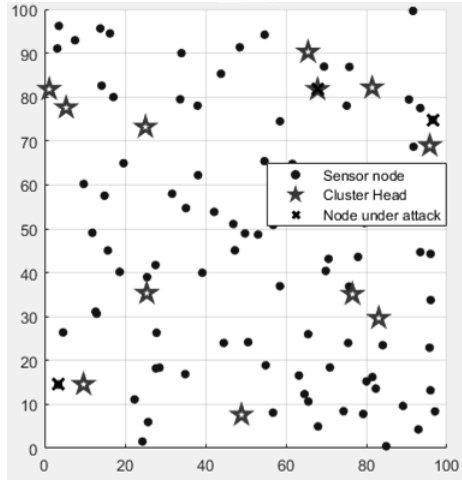


Fig. 6. Simulation of the LEACH protocol in MATLAB

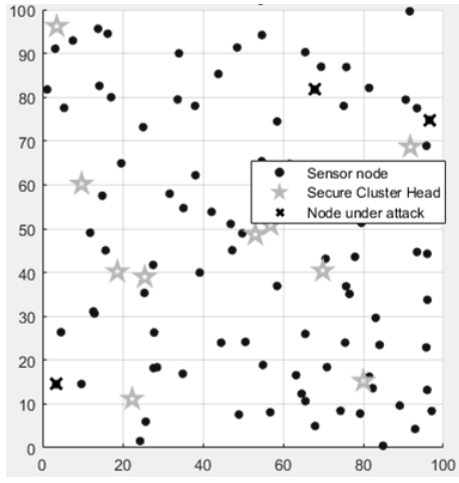


Fig. 7. Simulation of the fuzzy-based clustering in MATLAB

As classical LEACH relies on mathematical probability, it can randomly select a malicious node or a node with low energy as the head. In the graph in Fig.6, illustrating the simulation results, LEACH selected a malicious node as CH, marked as a crossed star. As shown in Fig.7, the proposed fuzzy-based scheme selected only secure nodes as cluster heads by

considering behavioral parameters and indirectly incorporating security indicators, thereby isolating malicious nodes and achieving a more secure clustering structure.

To ensures a comprehensive evaluation of efficiency and security aspects of the proposed clustering scheme, the performance of the two considered approaches was assessed by four metrics: network lifetime, energy consumption, latency, and robustness against attacks. The simulation results demonstrated that the proposed fuzzy scheme prolongs the network lifetime compared to standard LEACH. By penalizing nodes with low RE, the fuzzy scheme promotes more even energy load balancing in the network, that results in a slower decline in the network energy (Fig. 8). Here, the total network residual energy metric represents the aggregate remaining energy of all operational sensor nodes within the network measured at each successive simulation round. In a real scenario, each IoMT sensor node measures its own residual energy. These data are sent to the base station or to the cloud gateway, which aggregates these values from all nodes to compute the total network energy and monitor overall system status.

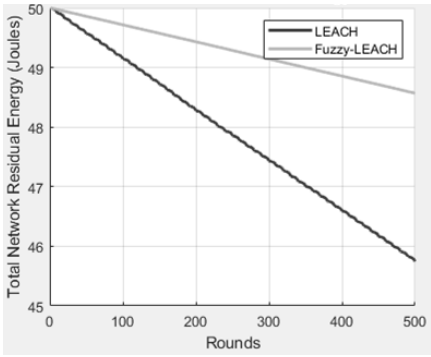


Fig. 8. Total network residual energy in MATLAB simulation

Latency was evaluated by tracking the average delay ratio of the selected CH. As the LEACH neglects the node behavior, it may select these compromised nodes as CHs, that results in latency spikes (fig.9). At the same time, the fuzzy-based scheme excludes nodes with high latency from the CH election.

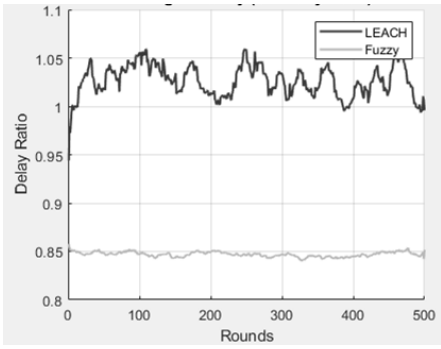


Fig. 9. Average latency in MATLAB simulation

The robustness testing showed that the proposed scheme achieved a near 100% avoidance accuracy; malicious nodes were systematically assigned a "Very Low" Chance Index and were successfully prevented from assuming the CH role in 500 rounds of simulation.

To quantitatively validate the resilience of the proposed fuzzy scheme against the attack scenario, a security analysis was performed based on standard classification metrics. The proposed scheme's capacity to isolate compromised nodes during the CH election phase was evaluated by computing the true positive rate (TPR) and the false positive rate (FPR).

The simulation results demonstrate that the proposed optimized fuzzy scheme achieves a high TPR of 100%, successfully identifying and isolating all simulated malicious nodes across the operational rounds. Furthermore, the scheme shows a low FPR of around 14% (fig. 10). Thus, the high classification accuracy confirms that integrating traffic-behavioral indicators with RE metrics via FL and GA provides a robust and secure clustering mechanism.

```
True Positives (TP)  : 3 (Attacks successfully blocked)
True Negatives (TN)  : 83 (Normal nodes allowed)
False Positives (FP) : 14 (Normal nodes erroneously blocke
False Negatives (FN) : 0 (Attacks missed by the system)
----------------------------------------------------------
Detection Rate (TPR) : 100.00%
False Positive Rate (FPR) :  14.43%
Overall Accuracy     :  86.00%
```

Fig. 10. Security validation metrics (confusion matrix)

In contrast to traditional trust models that require high computational overhead, the proposed scheme integrates trust management component into the FIS. In this scheme, trust is not maintained as a separate feature; it is computed and considered as the node's eligibility metric, the Chance Index. In case when multiple compromised nodes collude, the developed FIS may be deceived. The point is that defending against collusion requires a trust aggregation mechanism, which is beyond the scope of this inquiry. Moreover, the proposed system is developed for implementation in a stationary network topology, which is applicable for IoMT deployments. In highly dynamic environments, the proposed secure clustering scheme may have too high FPR, as it may consider benign moving nodes to be malicious ones.

VII. CONCLUSION

Ensuring security in WSNs is an essential task because these networks often operate in sensitive remote environments, such as military or healthcare applications, where data integrity and confidentiality are vital. Moreover, unauthorized access can lead to data breaches or system manipulation. Without adequate security measures, WSNs would be quite vulnerable to different types of attacks as implementing security protocols, algorithms and schemes for clustering helps protect the network from malicious threats. Artificial Intelligence has opened new horizons in efficient and intelligent clustering for WSN. These methods enable optimized decision-making, thus improving efficiency and security of such networks.

In this study, a secure clustering scheme for WSNs was proposed, using a fuzzy logic-based approach for the cluster head selection process. A fuzzy inference system was developed for evaluating the suitability of nodes for being elected as cluster heads based on several input parameters, thus enabling more adaptive decision-making mechanism in WSN environment. The developed fuzzy inference system was implemented and simulated in the MATLAB program for

evaluating its effectiveness. To improve the performance of the fuzzy inference system, a genetic algorithm-based optimization procedure was applied to fine-tune its membership functions and rule base. The developed FIS was validated, that confirmed its operability. The proposed in this study optimized secure clustering scheme can be applied in clustering process within IoMT, where data security is often a critical issue.

Future investigations will be focused on integration of additional security parameters with other optimization techniques to further extend the applicability of the proposed approach. Moreover, the proposed secure clustering scheme may be extended through integration with lightweight blockchain mechanisms, validation on a real-world testbed and adaption to mobile WSN to enhance practical applicability.

REFERENCES

[1] B. Rashid and M. H. Rehmani, "Applications of wireless sensor networks for urban areas: A survey," *Journal of Network and Computer Applications*, vol. 60, pp. 192–219, Jan. 2016, DOI: 10.1016/j.jnca.2015.09.008.

[2] F. Afroz and R. Braun, "Energy-efficient MAC protocols for wireless sensor networks: a survey," *International Journal of Sensor Networks*, vol. 32, no. 3, p. 150, 2020, DOI: 10.1504/ijsnet.2020.105563.

[3] Mansour Lmkaiti, Ibtissam Larhlimi, Maryem Lachgar, Houda Moudni and Hicham Mouncif "Framework for Intrusion Detection in IoT Networks: Dataset Design and Machine Learning Analysis", *Infocommunications Journal*, Vol. XVII, No 2, June 2025, pp. 61–71., DOI: 10.36244/ICJ.2025.2.8

[4] O. Olufemi Olakanmi and A. Dada, "Wireless Sensor Networks (WSNs): Security and Privacy Issues and Solutions," *Wireless Mesh Networks - Security, Architectures and Protocols*. IntechOpen, May 13, 2020. DOI: 10.5772/intechopen.84989.

[5] F. Azzedin and H. Albinali, "Security in Internet of Things: RPL Attacks Taxonomy," in *Proc. of the 5th International Conference on Future Networks and Distributed Systems*, pp. 820–825, Dec. 15, 2021. DOI: 10.1145/3508072.3512286.

[6] S. Khanam, I. B. Ahmedy, M. Y. Idna Idris, M. H. Jaward, and A. Q. Bin Md Sabri, "A Survey of Security Challenges, Attacks Taxonomy and Advanced Countermeasures in the Internet of Things," *IEEE Access*, vol. 8, pp. 219 709–219 743, 2020, DOI: 10.1109/access.2020.3037359.

[7] J. Kshirsagar and V. Sonaje, "Wireless Sensor Networks and the LEACH Protocol: Analysis of network using certain parameters for clustering and localization.," *International Journal of Computer Applications*, vol. 186, no. 41, pp. 1–6, Sep. 2024, DOI: 10.5120/ijca2024924003.

[8] V. K. Kumar and A. Khunteta, "Energy Efficient PEGASIS Routing Protocol for Wireless Sensor Networks," *2018 2nd International Conference on Micro-Electronics and Telecommunication Engineering (ICMETE)*. IEEE, pp. 91–95, Sep. 2018. DOI: 10.1109/icmete.2018.00031.

[9] H. B. Magar, P. Sharma, V. Philip, and A. S. Ubale, "Optimizing wireless sensor network performance: TEEN vs. LEACH analysis," *AIP Conference Proceedings*, vol. 3222. AIP Publishing, p. 050003, 2024. DOI: 10.1063/5.0228211.

[10] M. Wang, S. Wang, and B. Zhang, "APTEEN routing protocol optimization in wireless sensor networks based on combination of genetic algorithms and fruit fly optimization algorithm," *Ad Hoc Networks*, vol. 102, p. 102 138, May 2020, DOI: 10.1016/j.adhoc.2020.102138.

[11] P. Subbulakshmi and M. Prakash, "Mitigating eavesdropping by using fuzzy based MDPOP-Q learning approach and multilevel Stackelberg game theoretic approach in wireless CRN," *Cognitive Systems Research*, vol. 52, pp. 853–861, Dec. 2018, DOI: 10.1016/j.cogsys.2018.09.021.

[12] J. Suhonen, M. Kohvakka, V. Kaseva, T. D. Hämäläinen, and M. Hännikäinen, "Low-Power Wireless Sensor Network Platforms," *Handbook of Signal Processing Systems*. Springer New York, pp. 381–419, 2013. DOI: 10.1007/978-1-4614-6859-2_13.

[13] A. Hamzah, M. Shurman, O. Al-Jarrah, and E. Taqieddin, "Energy-Efficient Fuzzy-Logic-Based Clustering Technique for Hierarchical Routing Protocols in Wireless Sensor Networks," *Sensors*, vol. 19, no. 3, p. 561, Jan. 2019, DOI: 10.3390/s19030561.

[14] O. Semenova, N. Kryvinska, A. Semenov, A. Rudyk, and A. Dzhus, "Intelligent Admission Control in Wireless Networks of IoT," *International Journal of Control, Automation and Systems*, vol. 23, no. 4, pp. 1262–1270, Apr. 2025, DOI: 10.1007/s12555-024-0603-z.

[15] Olivér Hornyák, "Intelligent Intrusion Detection Systems – A comprehensive overview of applicable AI Methods with a Focus on IoT Security", *Infocommunications Journal*, Special Issue on AI Transformation, 2025, pp. 61–76, DOI: 10.36244/ICJ.2025.5.8

[16] Mohd Aaqib Lone, Owais Mujtaba Khanday, and Szilveszter Kovács, "Implementation Guidelines for Ethologically Inspired Fuzzy Behaviour-Based Systems", *Infocommunications Journal*, Vol. XVI, No 3, September 2024, pp. 43–56., DOI: 10.36244/ICJ.2024.3.4

[17] P. Schaffer, K. Farkas, Á. Horváth, T. Holczer, L. Buttyán: "Secure and Reliable Clustering in Wireless Sensor Networks: A Critical Survey", *Elsevier Computer Networks*, vol. 56, no. 11, pp. 2726–2741, July 2012, DOI: 10.1016/j.comnet.2012.03.021.

[18] R. Elavarasan and K. Chitra, "An efficient fuzziness based contiguous node refining scheme with cross-layer routing path in WSN," *Peer-to-Peer Networking and Applications*, vol. 13, no. 6, pp. 2099–2111, Jan. 2020, DOI: 10.1007/s12083-019-00825-0.

[19] R. K. Poluru, M. P. K. Reddy, S. M. Basha, R. Patan, and S. Kallam, "Enhanced Adaptive Distributed Energy-Efficient Clustering (EADEEC) for Wireless Sensor Networks," Recent Advances in *Computer Science and Communications*, vol. 13, no. 2, pp. 168–172, June 2020, DOI: 10.2174/2213275912666190404162447.

[20] P. Xie, M. Lv, and J. Zhao, "An improved energy-low clustering hierarchy protocol based on ensemble algorithm," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 7, Nov. 2019, https://doi.org/10.1002/cpe.5575.

[21] C. Zhu, S. Wu, G. Han, L. Shu, and H. Wu, "A Tree-Cluster-Based Data-Gathering Algorithm for Industrial WSNs With a Mobile Sink," *IEEE Access*, vol. 3, pp. 381–396, 2015, DOI: 10.1109/access.2015.2424452.

[22] A. Daniel, K. M. Balamurugan, R. Vijay, and K. Arjun, "Energy Aware Clustering with Multihop Routing Algorithm for Wireless Sensor Networks," *Intelligent Automation and Soft Computing*, vol. 29, no. 1, pp. 233–246, 2021, DOI: 10.32604/iasc.2021.016405.

[23] H. Li and J. Liu, "Double Cluster Based Energy Efficient Routing Protocol for Wireless Sensor Network," *International Journal of Wireless Information Networks*, vol. 23, no. 1, pp. 40– 48, Mar. 2016, DOI: 10.1007/s10776-016-0300-9.

[24] B. Balakrishnan and S. Balachandran, "FLECH: Fuzzy Logic Based Energy Efficient Clustering Hierarchy for Nonuniform Wireless Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 2017, pp. 1–13, 2017, DOI: 10.1155/2017/1214720.

[25] S. M. Bozorgi and A. M. Bidgoli, "HEEC: a hybrid unequal energy efficient clustering for wireless sensor networks," *Wireless Networks*, vol. 25, no. 8, pp. 4751–4772, May 2018 DOI: 10.1007/s11276-018-1744-x.

[26] A. Jain and A. K. Goel, "Energy Efficient Fuzzy Routing Protocol for Wireless Sensor Networks," *Wireless Personal Communications*, vol. 110, no. 3, pp. 1459–1474, Oct. 2019, DOI: 10.1007/s11277-019-06795-z.

[27] K. Shaukat *et al.*, "MAC Protocols 802.11: A Comparative Study of Throughput Analysis and Improved LEACH," *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*. IEEE, pp. 421–426, June 2020. DOI: 10.1109/ecti-con49241.2020.9158097.

[28] A. Abdulwahid and M. Salih, "Wireless Sensor Networks Applications, Challenges, and Security Requirements," in *Proc. of 2nd International Multi-Disciplinary Conference Theme: Integrated Sciences and Technologies*, IMDC-IST 2021, 7-9 September 2021, Sakarya, Turkey. EAI, 2022. DOI: 10.4108/eai.7-9-2021.2314823.

[29] R. Saatchi, "Fuzzy Logic Concepts, Developments and Implementation," *Information*, vol. 15, no. 10, p. 656, Oct. 2024, DOI: 10.3390/info15100656.

[30] MATLAB Fuzzy Logic Toolbox Documentation. [Online]. Available: https://www.mathworks.com/help/fuzzy/building-systems-with-fuzzy-logic-toolbox-software.html

[31] K. Svandova and Z. Smutny, "Internet of Medical Things Security Frameworks for Risk Assessment and Management: A Scoping Review," *Journal of Multidisciplinary Healthcare*, vol. Volume 17, pp. 2281–2301, May 2024, DOI: 10.2147/jmdh.s459987.

Vincent Karovič is a graduate of the Faculty of Management, Comenius University in Bratislava. He has been working at the Faculty of Management, Comenius University, since 2017, where he teaches courses related to cybersecurity and IT management. His research focuses primarily on information systems and cybersecurity. Since his doctoral studies, he has specialized in the management of e-government as an extension of IT management research in the public sector. He is a member of the Department of Information Management and Business Systems and currently serves as the Vice-Dean for IT at the Faculty of Management, Comenius University.

Olena Semenova received the MSc degree in Radio Engineering in 2003 and the Ph.D. degree in Computer Systems and Elements in 2007 from Vinnytsia National Technical University, Vinnytsia, Ukraine. She is currently an Associate Professor in the Department of Infocommunication Systems and Technologies at Vinnytsia National Technical University. Her research interests include telecommunication networks and artificial intelligence. She has authored more than 180 scientific and academic publications.

Maksym Prytula received the MSc degree in Radio Engineering in 2007 and the Ph.D. degree in Radio measuring instruments in 2021 from Vinnytsia National Technical University, Vinnytsia, Ukraine. He is currently an Associate Professor in the Department of Information Radioelectronic Technologies and Systems at Vinnytsia National Technical University. His areas of research interest include magnetic sensors and magnetic field measurements. He has authored more than 80 scientific and academic publications.

Volodymyr Martyniuk received the BSc degree in Telecomunications in 2009 and the MSc degree in Telecomminication and Radiotechnics in 2010 from Vinnytsia National Technical University. He is currently a postgraduate student and works as a JavaScript Technical Lead at Playtika Holding Corp. His research interests include telecommunication networks, artificial intelligence, and programming.

Vladyslav Kuzniak received the MSc degree in Medicine in 2022 from Vinnytsia National Medical University and the MSc degree in Computer Science in 2024 from Vinnytsia National Technical University. He is currently a postgraduate student and works as a Head of AI Department at LITSLINK LLC. His research interests include telecommunication networks, artificial intelligence, molecular biology, and medicine.

Synchronized Real-time Communication In The Eclipse Arrowhead Framework

Attila Frankó[‡&], Gergely Hollósi[‡], Dániel Ficzere[‡], and Pál Varga[‡]

Abstract—**Real-time, deterministic communication is a common requirement in most time-critical industrial applications. Due to the increasing complexity, interoperability of such systems must be handled by using appropriate design principles and convenient tools that enhance and ease composability. The Eclipse Arrowhead Framework aims to be a basis for such industrial Cyber-Physical System of Systems by providing a standard for services and enabling interoperability between systems and services. However, precise timing requirements cannot be handled on a service basis as they are closely related to hardware capabilities and the structure of the physical underlying network. This paper addresses this by proposing an extension of the Arrowhead framework to bridge synchronization between instances to the service level without losing any abstraction, thus preserving fundamental achievements of the Arrowhead, such as late binding and loose coupling.**

Index Terms—**real-time, synchronization, arrowhead, ptp, cyber-physical systems, system of systems, clock domains**

I. INTRODUCTION

In the dynamic landscape of industrial services and applications, time synchronization emerges as a critical factor influencing the efficiency, reliability, and interoperability of diverse processes. As industries undergo digital transformation and embrace cutting-edge technologies, the orchestration of interconnected Cyber-Physical Systems of Systems (CPSoS) becomes increasingly complex, necessitating precise timekeeping for seamless coordination. Accurate time synchronization is integral to successfully executing numerous industrial functions, ranging from manufacturing processes to data communication and control systems. In manufacturing, synchronized time ensures the harmonious collaboration of machines and devices, optimizing production workflows and minimizing delays. Furthermore, in distributed control systems and sensor networks, precise time synchronization is paramount for maintaining temporal coherence, enabling accurate data correlation, and enhancing overall system resilience.

Maintaining precise time synchronization becomes paramount in cloud environments, where diverse applications, services, and resources interact across geographically dispersed data centers. One of the key areas where time synchronization plays a pivotal role is in distributed computing and virtualization. Cloud services often rely on virtual machines and containers distributed across multiple servers. Accurate time synchronization is essential for these virtualized instances to operate cohesively, facilitating streamlined communication and synchronization of activities.

One of the key motivations for prioritizing time synchronization within an Industrial IoT (IIoT) framework lies in its role in enabling predictable and deterministic behaviors in industrial systems. From manufacturing processes to control systems, synchronized time ensures that actions occur precisely when needed, contributing to the overall predictability and efficiency of industrial operations. In this work, we use the Arrowhead IIoT framework for its capabilities, such as loose coupling, interoperability, and widespread research support, facilitating the practical application of new results. The Arrowhead framework's commitment to time synchronization is also evident in its support for distributed architectures and service-oriented paradigms. As industrial systems evolve toward decentralized and interconnected models, precise timekeeping becomes indispensable for maintaining synchronization among distributed entities, thereby enhancing system resilience and responsiveness.

Within our paper, we introduce the architecture and functioning of the synchronized real-time communication embedded in the Arrowhead Framework. Additionally, the paper delineates the primary requirements for time synchronization functionality, generally for Service Oriented Architectures (SoA) and also specifically within the Arrowhead framework, accompanied by a case study focusing on scheduled execution and synchronized service orchestration.

The paper structured as follows: Section II introduces the major concepts and elements of the Arrowhead framework and PTP protocol as the primary enabler of precise time synchronization. Section III details the motivation and requirements for achieving synchronized communication between services in Arrowhead and also presents the proposed architectural extension. Section IV shows an application example of the proposed solution while section V concludes the paper.

II. RELATED WORKS

A. Arrowhead Framework

We outline the fundamental principles of the Arrowhead project to understand the key guidelines and motivations behind the development of its time synchronization architecture. The Arrowhead project is guided by three core principles, which are as follows:

[‡] Dept. of Telecommunications and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary
[&] IIoT & AI Division, AITIA International Inc., Budapest, Hungary
corr.: attila.franko@vik.bme.hu

DOI: 10.36244/ICJ.2026.2.11

1) Local information exchange: It recognizes that automation and IIoT systems often come with specific requirements, resulting in ”local” information exchanges. In such settings, systems are in close proximity to each other, forming a closed System of Systems architecture. Consequently, Arrowhead emphasizes the importance of addressing the ”closedness” of industrial applications. It operates under the assumption that various application systems and devices can only operate within their own local or closed automation network, known as the ”local Arrowhead network” or ”cloud.” Essentially, an Arrowhead Local Cloud (LC) is a System of Systems, where a node, referred to as a ”system,” within the LC can itself be a complex system, such as a machine, engaging in information exchange with other ”systems” like controllers.

2) Service-Oriented Architecture: Arrowhead advocates the utilization of Service-Oriented Architecture (SOA) within the realm of automation. Within an Arrowhead LC, multiple application systems communicate and provide various functional services to each other. In Arrowhead’s terminology, this is represented as one application system serving as a Service Provider, offering services to another application system, and becoming a Service Consumer. In this context, Arrowhead incorporates all three essential aspects of SOA: (i) loose coupling, (ii) late binding, and (iii) the ability to look up partners at runtime. The definition of a Service outlines the invokable functionality and its interface implementation [9].

Loose coupling Strict, hardwired connections between entities are strictly prohibited.

Late binding Services are dynamically discoverable at runtime by systems that need them rather than being predefined.

Self-containment and modularity Services are self-contained and modular, offering a set of logically cohesive interfaces (e.g., an Application Programming Interface - API) that implement similar functionalities.

Interoperability Systems utilizing different platforms and programming languages should be able to communicate with each other using pre-defined implementations of various services.

Generally, an application system is an execution logic deployed on a hosting device and is part of a cyber-physical system with real-world sensing and actuation capabilities. Arrowhead does not make assumptions about what can constitute an application system; it can be as simple as a single sensor mote or as complex as an entire smart city. The emphasis is on the fact that each application system provides and consumes services from other systems within the LC, with these information exchanges governed by the Arrowhead Core Systems [22].

3) Core Systems: Arrowhead acknowledges that a System of Systems cannot operate entirely in a decentralized manner due to operational and business considerations. To address this, Arrowhead offers several Core Systems designed to control, facilitate, and assist service connections within an LC. There are three mandatory core systems:

Service Registry Service Providers use the Service Registry to register the services they offer and their network addresses, facilitating loose coupling and partner lookup.

Authorization System This system is responsible for managing access permissions, and it issues authorization tokens that can only be decrypted by the designated Service Provider System. These tokens contain information about the consumer system, the allowed service interface, and the duration of access.

Orchestrator The Orchestrator supports the establishment of connection rules, matching service consumers with suitable service providers. It relies on the Orchestration Store, where LC operators can define orchestration rules.

Arrowhead also provides non-mandatory supporting core systems to enhance the central functionalities of application systems. These systems assist with tasks related to event handling, protocol translation, Quality of Service management, and more [11]. Fig. 1 shows the mandatory core systems and supporting core systems for inter-cloud communications implementing the IIoT reference architecture of the Industrial Internet Committee.

B. Time synchronization In Industrial Automation Systems

In recent years, the emergence of CPSoS led us to highly automated manufacturing with autonomous production lines and shop floors, or even lights-out, smart factories that minimize the required human interaction during the process [10], [19]. These systems usually involve mission-and time-critical requirements in order to control and monitor autonomous devices (vehicles, robots, etc.) precisely [16], [20]. In such complex systems, these components usually interact with services outside of the mission-critical part and take information from outside of the system as they are integral parts of a global supply chain network [7], [12].

Achieving real-time or deterministic communication and execution has always been a key factor for time-critical applications. Nowadays, due to the increasing number of cyber-physical-systems-driven production systems in smart industry [21] this topic is more crucial than ever. Generally, these systems often have to meet real-time requirements or rely on accurate time synchronization to coordinate the actions of multiple machines precisely.

In the last few decades, great efforts have been made in order to provide different solutions that can address issues related to real-time requirements such as low latency communication, time synchronicity, scheduling, priority handling (known as traffic shaping), fault tolerance, especially in challenging environemts [14], [18]. Recently, the existing, widely used, and industrially accepted solutions have been standardized and grouped under the umbrella term Time-sensitive Networking (TSN) [3]. TSN is really a set of various standards that aims to be an enabler of such systems, which have to meet one or more aspects of the aforementioned real-time requirements [2]. In this paper, we mainly focus on time synchronization, including the underlying technologies and protocols where necessary.

Precision Time Protocol (PTP) or IEEE 1588 is a protocol that is used to provide time synchronicity between different clocks in a network [4]. PTP is one of the most important protocols of TSN as part of the IEEE 802.1AS standard, which restricts the usage of PTP to a specific profile often referred

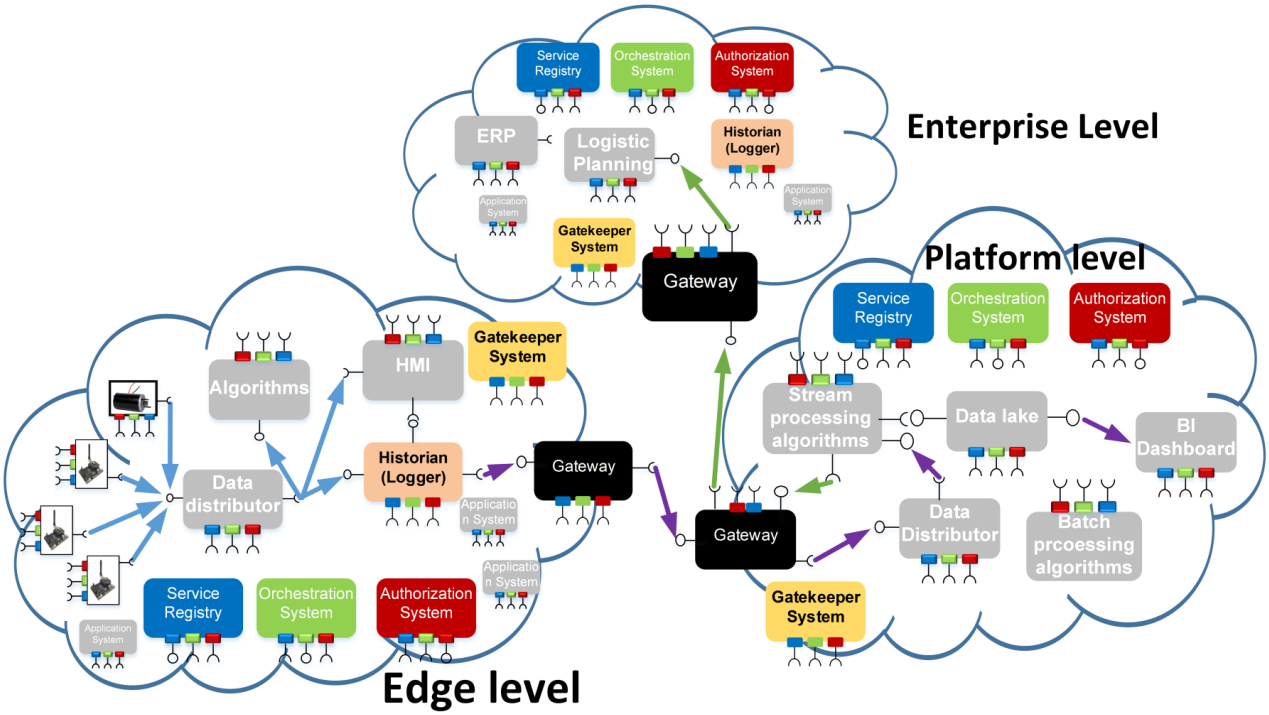


Fig. 1: The IIC (Industry IoT Consortium) IIoT Reference Architecture mapped in Arrowhead presenting its core systems and related systems [11]

to as gPTP (where 'g' stands for general(ized)) defined by the standard [5].

IEEE 1588 strictly defines the clock synchronization model, the PTP entities, the clock types, and the messaging itself. However, we only discuss those parts that are mandatory for presenting the proposed concept. In a computer network, PTP can synchronize different PTP *instances*, which can be parts of one or multiple PTP nodes. Practically, instances are dedicated physical clocks of a Network Interface Card (NIC), while a node can be a server machine or a switch. A network to be synchronized does not require a consisting of only PTP nodes. Moreover, switches do not even have to be PTP-capable ones; in practical cases, they are usually demanded, while in TSN, they are mandatory.

To achieve synchronization, a grandmaster (GM) clock is selected from all PTP instances and will be the reference for all clocks in the network. PTP allows different synchronization methods (protocols) to be used: End-to-end (E2E) and peer-to-peer (P2P) over multiple hops, but each case can be simplified to a master-slave synchronization of two neighboring (i.e., they are connected directly with a link) instances as shown in Figure 2. The core method involves several messages that include the timestamp of receiving and transmitting a specific message; the slave can calculate the link delay and the time offset between the two instances – if the offset is smaller than a certain value, then only the frequency of the clock is trimmed. At the application level, a clock instance can be accessed directly or via the operating system (OS). Thus, any application is able to use any precisely synchronized clock of the node.

Since one PTP node can be a part of multiple networks, thus PTP instances of a node can synchronize to different clocks as well: a collection of instances synchronized to the same clock (and part of the same network) is a clock domain. It is often desired in industrial environments that one node can use timing from different clock domains for its operations. Moreover, according to (the current status of) IEC/IEEE 60802 TSN Profile for Industrial Automation, the usage of more than one clock domain is the standard in industrial use-cases in order to use Working Clock and Global Time domains and also for hot-standby operations [13]. Our proposed solution also supports the usage of multiple clock domains to be aligned with the industrial requirements.

[6] provides a comprehensive survey on Time-Sensitive Networking within the automotive domain, identifying key research opportunities in the integration of deterministic communication with higher-level software architectures. Our work extends this vision by proposing a service-oriented bridge that enables the dynamic orchestration of these timing-critical features within the Eclipse Arrowhead framework, a layer that is often treated as a static configuration in current automotive TSN implementations.

To provide a structured overview of the current landscape, Table I compares existing and prevalent industrial synchronization approaches based on the taxonomy and requirements identified in the literature, specifically focusing on the trade-offs discussed by [17]. It is worth noting that they are not synchronization protocols per se, but architectures and frameworks

TABLE I
COMPARSION OF COMMUNICATION ARCHITECTURES FOR SYNCHRONIZATION IN THE IIOT LANDSCAPE [17], [6], [8]

	SDN-TSN	OPC UA PubSub	FIWARE
Layer Focus	Data Link (L2)	Middleware (L4-L7)	Application (SOA)
Synchronization Precision	Nanosecond	Micro/Nanosecond	Millisecond
Time Source	HW Grandmaster	HW/SW Hybrid	Network (NTP)
Determinism	Gating (TAS)	Priority-based	Best-effort
Coupling	Very Tight	Tight (Static)	Loose
Configuration Phase	Design-time	Semi-static	Runtime / Dynamic

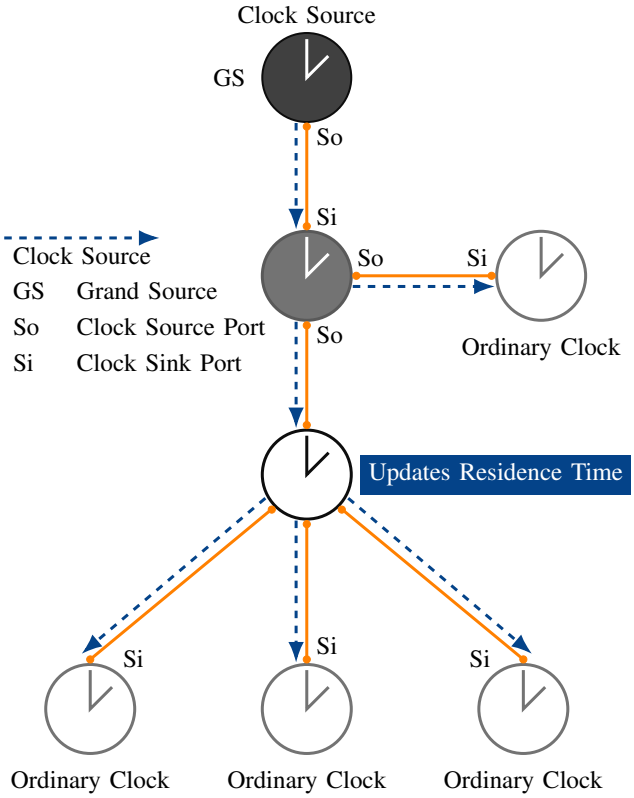


Fig. 2: Different types of PTP clocks and how shared time getting distributed – based on [1]

that use PTP or NTP for achieving synchronicity. The analysis covers a spectrum from low-level network protocols to high-level cloud-based IoT frameworks. This comparison highlights a significant r esearch g ap: w hile h ardware-centric solutions like SDN-based TSN [17] offer nanosecond precision, they often impose rigid, design-time configurations. T hese solutions utilize PTP as the underlying protocol that implements synchronization, so they differ in the level of abstraction and, most importantly, in their focus. Conversely, service-oriented frameworks like FIWARE [8] prioritize flexibility a nd loose coupling but fail to provide the deterministic timing necessary for mission-critical industrial operations, as it uses NTP, which provides only millisecond precision. By evaluating these dimensions—precision, determinism, and coupling strategy- it becomes evident that a middleware-independent, service-level synchronization bridge is required to maintain the dynamic nature of System-of-Systems architectures without sacrificing real-time performance.

III. SERVICE CONSUMPTION IN A SYNCHRONIZED MANNER WITHIN ARROWHEAD

A. Requirements and motivation

Generally, IIoT and SoS frameworks and platforms offer different functions to enhance and ease the design and implementation of microservice-based solutions. Usually, these tools are promoted as major enablers for time-critical applications; however, in most cases, only real-time response times are targeted, which are achieved as per service response time and fast communication to minimize the delay. Nonetheless, since synchronicity requires special hardware resources, including end devices and network devices as well, it is not addressed on a service basis but on a device basis (such as Network Interface Cards or clock instances).

Arrowhead is unique regarding its approach to interoperability and interchangeability. However, service consumption is a focal point as it is a key element of modern system architecture. In order to enable real-time, deterministic communication between Arrowhead-compliant services – including precise, scheduled execution of different tasks provided by services – the current architecture has to be expanded further; it is worth noting that scheduling here does not mean chaining calls based on a predefined order, as it is done very similarly by supplementary core system Workflow Choreographer [15]. It means only precise execution of a task, i.e., invocation and running a service at the exact time. Nonetheless, in industrial systems, automated, flexible workflow execution is often required to be performed with precise, synchronized timing. Thus, the Choreographer and our proposed solution can jointly make systems meet these requirements.

In [15], authors provide a real use case in order to demonstrate the need for workflow management and apply their solution to that. The use case includes a generalization of a common problem of moving and packing objects on shop floors or production lines. This task requires not only workflow management but precise, synchronized execution as well, since in a real environment, each step has to be performed just in time, especially if we extend the number of actors that have to execute actions cooperatively. Consequently, all devices involved in a time-critical process have to be synchronized to enable precise, scheduled execution.

A fundamental issue is the control plane being defined on the service level, including the core systems of the local cloud, while synchronicity exists per device – or more precisely, per instance – level. Therefore, the main issue here is not achieving synchronicity between services within the Local Cloud, as it would highly restrict service abstractions, but using the already existing time synchronicity between devices that run the services. It is crucial to avoid involving concrete devices in the design of systems. Therefore, we have to transform the timing capabilities to the service level.

Another issue raised by industrial use cases is the problem of multiple clock domains. A naive way to achieve synchronicity within a local cloud could be the enforcement of synchronicity between all hosts that provide or consume a service in a local cloud. However, this approach assumes using only one clock domain as it implicitly requires the node to be synchronized with the same grandmaster. Our proposed solution follows a different concept that does not restrict the number of used clock domains. Thus, the synchronicity of the instances does not have to be global in the local cloud. Instead, Arrowhead is fully aware of the existence of different clock domains within the local cloud and is able to take this information into consideration during the orchestration process.

B. Arrowhead TimeSync Instance and Time Sync Relay

The bridging of synchronicity to the control plane of Arrowhead is handled by a Time Synchronization Relay (TSR) supporting core system, as it can be seen in Figure 4 and a special consumer service profile called Arrowhead TimeSync Instance (ATI) as seen in Figure 3.

The ATI is a special service in systems that enables synchronization for certain services within a system. It operates as a scheduler proxy for any service that runs on the same PTP instance as the ATI. Since the synchronization is handled by the PTP protocol itself, the ATI has access to a physical clock that is synchronized to one or more other PTP instances in the Local Cloud. Generally, there are no guarantees that a service can use any physical clock of the machine that runs it – e.g., if the service runs in an isolated environment such as a container, it might not be able to use the clock. However, an ATI must always have access to the desired PTP instance. It is worth emphasizing that since devices can have multiple PTP instances that are synchronized to different clock domains, it is possible to run multiple ATI-s on the same device to schedule tasks by different clocks or have dedicated ATI-s for different services (see below).

With that approach, service invocation can be planned beforehand since ATI can schedule it precisely: we will call this service of an ATI a Scheduled Execution (SCE). It is important to emphasize that a service can be invoked directly via Arrowhead as a regular service, even if it offers real-time execution, which can be soft real-time or hard real-time. These characteristics of ATI depend upon the implementation of ATI, as calling the desired service might cause delays during invocation. To meet soft real-time requirements, ATI can be realized as a general software that can call other services locally via protocols using shared memory or inter-process calls (IPC). In order to provide hard real-time execution with practically zero delay, ATI must be implemented as an integral part of any real-time-capable service (using it as a dedicated library). In such a way, the service has its own dedicated ATI, so there might be different ATI-s using the same PTP instance. It is worth noting that the soft real-time approaches enable legacy services to offer real-time execution as they don't have to be bundled with a dedicated ATI.

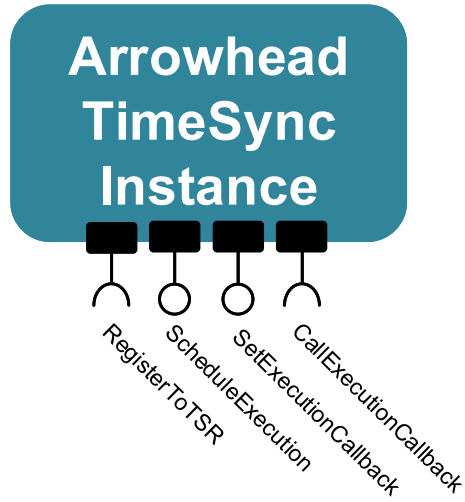


Fig. 3: Overview of Arrowhead TimeSync Instance

Time Sync Relay provides services for systems to ensure time-critical execution. Services of a system can register themselves at any time to advertise their SE capability. However, these services might offer non-scheduled execution as well – it depends on the service itself. When a service registers itself to the TSR, it has two options: (i) it must declare its ATI explicitly, thus enabling legacy applications to use external ATI for soft real-time scheduling or (ii) if they have internal ATI for hard real-time execution, then it must be determined automatically (e.g., domain name matching, etc.) so if an execution has to be scheduled for the service, the TSR knows which ATI should be addressed with the scheduling request (SCR). This implies that a SE-capable service must always know the identity of its ATI during its lifecycle. It is worth noting that despite the implied local hard coupling, this approach does not violate the overall philosophy and composability of the Arrowhead. From an orchestration perspective, the ATI is transparent; only the running service must know its ATI since they are coupled by the fact of using the same PTP instance.

Nevertheless, this is still not sufficient for SE since it does not provide any guarantees of two services being synchronized. It technically means that their corresponding PTP instances are synchronized to the same grandmaster, i.e., they are in the same PTP domain. The root of the problem is that two instances only know the grandmaster, to which they are synchronized, and real-time services usually do not require additional knowledge as they run on the same physical network by design. It is still true; however, during the orchestration

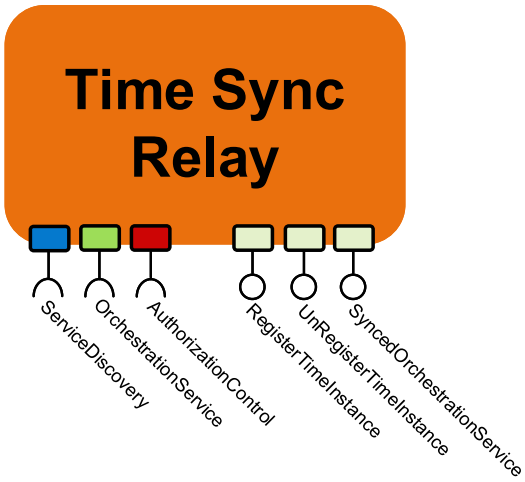


Fig. 4: Overview of Time Sync Relay

process, it is not revealed if a provider service runs on the same physical network as the consumer. Due to this, the orchestration process must offer services that run on the same physical network as the consumer. To address this issue, during registration, an ATI must always declare the identifier o f the PTP domain in which it is located. Thus, two services can determine whether they are synchronized to the same GM. An overview of the concept of ATI-s can be found in Figure 5.

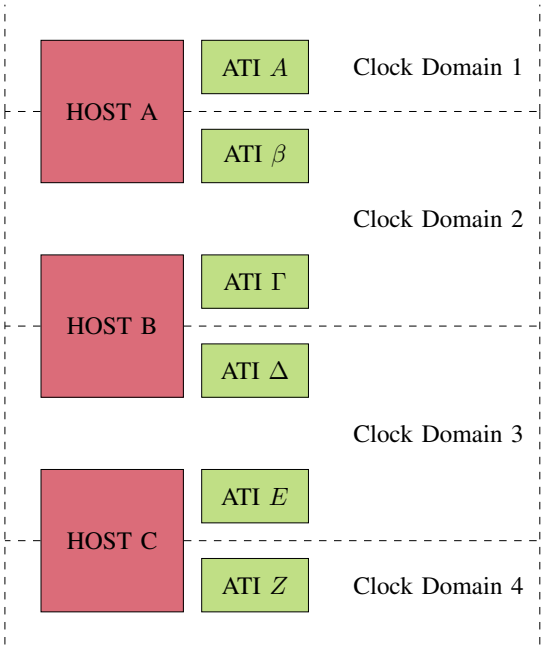


Fig. 5: Multiple hosts providing services while synchronized to different clock domains by multiple ATI-s

IV. Case study: Scheduled Execution and Synchronized Service Orchestration

In this section, we present an example of a Scheduled Execution (SE) and a Synchronized Service Orchestration (SSO), too, as they can be seen in Figure 6. This case shows a TSR and two real-time services with the corresponding ATIs in one LC. The steps of registration of a real-time service will be shown, but mandatory functions of an LC, such as registering a service to the Service Registry and setting authorization rules, will not. Moreover, the synchronization via the PTP process is also omitted as it is well-defined in the corresponding standard [4], so all PTP instances are assumed to be already synchronized to a common GM.

The example shows an abstract SE and SSO. However, it is the same for any configuration, including the production line use case mentioned in subsection III-A. It goes as follows:

1) The ATI of the ”to-be-provider” service registers itself to the TSR and also declares the PTP Domain ID to which it is synchronized.
2) The ”to-be-consumer” service requests regular orchestration with the desired flags and properties from the orchestrator.
3) The orchestrator returns the list of available services, such as for regular orchestration.
4) The ”to-be-consumer” service requests special orchestration for real-time services (that share a common clock domain) from the TSR based on the orchestrator’s response. The service calls the ”SyncedOrchestrationService“ endpoint of the TSR, passing the service description of the service to be consumed.
5) The TSR finds the corresponding ATI based on the request (service, clock domain) by applying different policies (e.g., URL matching) and methods and returns them to the caller. Now, it is possible for the caller to initiate an SSO directly with the provider service.
6) To initiate an SE, the consumer sends an SE request to the ATI, including the service definition of the service to be scheduled and other parameters (optionally the callback URL) by calling the ”ScheduleExecution“ service.
7) The ATI schedules the given call as it is defined in the request and then invokes the target service at the scheduled time.
8) Optionally, the provider service can return data (depending on its functions and service definition) to ATI.
9) The ATI forwards the return data to the consumer system via the preconfigured callback service ”CallExecutionCallback“.

V. Conclusions

Due to the ever-growing complexity of industrial systems, design principles address composability and interoperability issues between the components of a system. Arrowhead framework is specifically built to support these principles, enabling all of its users to work with a common and unified approach. However, these principles operate on a service basis, whereas specific industrial requirements, such as real-time, deterministic communications, function on a different level. This discrepancy makes it impossible to meet such demands solely through service abstractions.

This paper presented an approach to extend the current capabilities of Arrowhead to bridge time synchronicity achieved by

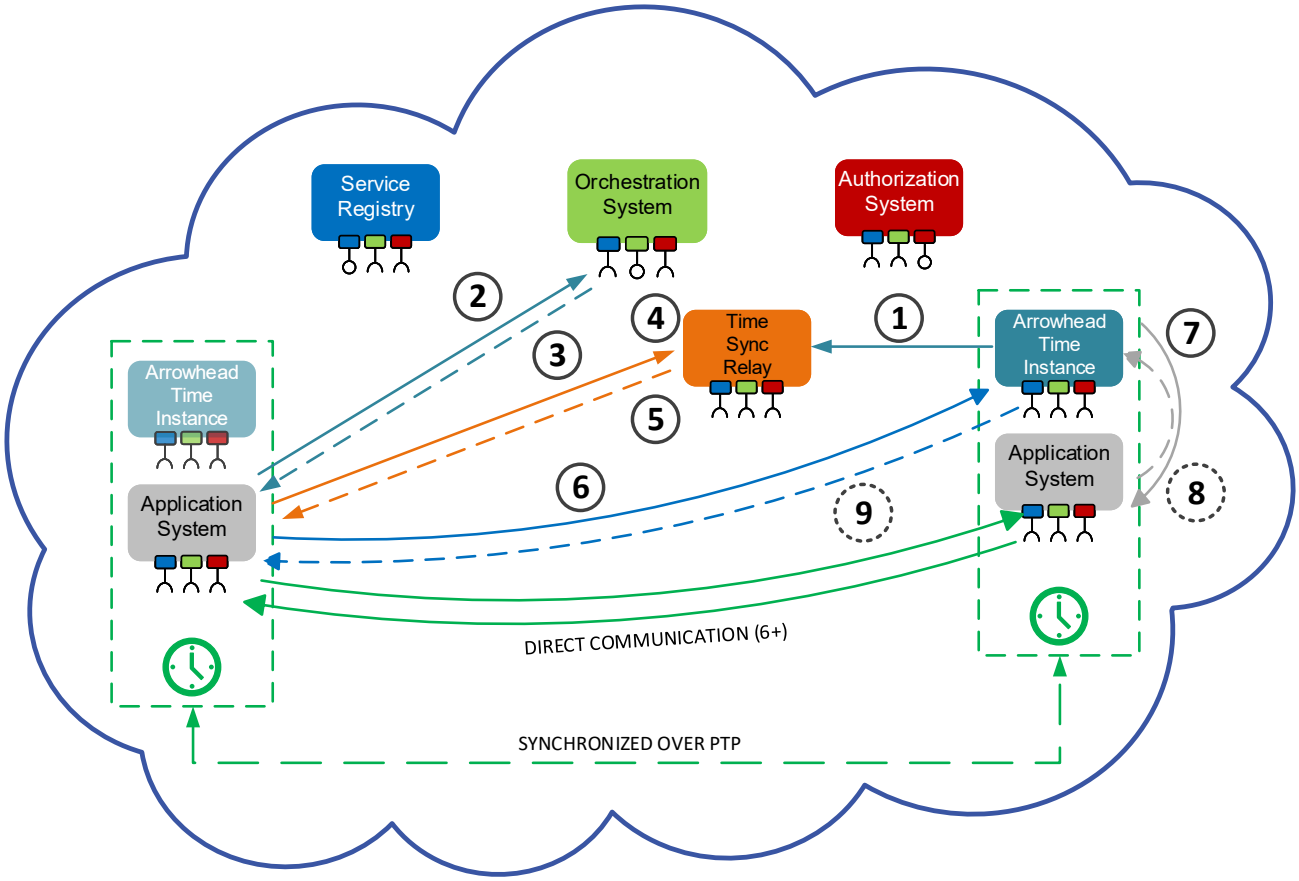


Fig. 6: Overview of an SE and SSO in a Local Cloud

PTP to service level, enabling loosely coupled CPSoS-es with precise, real-time communication and execution requirements. The overall solution extends the orchestrating capability of the Arrowhead with the help of Time Synchronization Relays to enable matchmaking that takes the time synchronicity of the services into account. Moreover, the Arrowhead Timesync Instance was introduced to provide a convenient way for proxying sync information to service level and also scheduled requests to services. Compared to the synchronization architectures summarized in Table I, the proposed approach combines deterministic PTP-based synchronization with the flexibility of a service-oriented architecture, eliminating the need for rigid design-time or semi-static configurations while preserving loose coupling. Furthermore, by supporting run-time orchestration and multiple clock domains, the proposed solution enables dynamic and interoperable industrial CPSoS deployments without compromising real-time communication and execution requirements.

REFERENCES

[1] Aos-cx 10.14.xxxx fundamentals guide, chapter 9: Ptp. https://arubanetworking.hpe.com/techdocs/AOS-CX/10.14/PDF/fundamentals_8100-83xx-9300-10000.pdf. [Online; accessed 23-April-2026].

[2] A real-time ethernet solution available in an open ieee 802.1 standard. https://tsn-network.com. [Online; accessed 1-Februar-2024].

[3] A deterministic ethernet solution based on time sensitive networking (tsn). https://www.ddc-web.com/resources/FileManager/dbi/Whitepapers/TSN%20White%20Paper.pdf, 2020. [Online; accessed 13-April-2026].

[4] IEEE standard for a precision clock synchronization protocol for networked measurement and control systems. *IEEE Std 1588-2019 (Revision of IEEE Std 1588-2008), pages 1–499, 2020.*

[5] IEEE standard for local and metropolitan area networks–timing and synchronization for time-sensitive applications. *IEEE Std 802.1AS-2020 (Revision of IEEE Std 802.1AS-2011)*, pages 1–421, 2020.

[6] Mohammad Ashjaei, Lucia Lo Bello, Masoud Daneshtalab, Gaetano Patti, Sergio Saponara, and Saad Mubeen. Time-sensitive networking in automotive embedded systems: State of the art and research opportunities. *Journal of Systems Architecture*, 117:102137, 2021.

[7] Pradeep Bedi, Kamal Upreti, Anand Singh Rajawat, Rabindra Nath Shaw, and Ankush Ghosh. Impact analysis of industry 4.0 on realtime smart production planning and supply chain management. In *2021 IEEE 4th International Conference on Computing, Power and Communication Technologies (GUCON)*, pages 1–6, 2021.

[8] Flavio Cirillo, Gürkan Solmaz, Everton Luís Berz, Martin Bauer, Bin Cheng, and Ernő Kovacs. A standard-based open source iot platform: Fiware. *IEEE Internet of Things Magazine*, 2(3):12–18, 2019.

[9] Jerker Delsing, editor. IoT Automation. CRC Press, February 2017. [10] Isa Gerekli, Tarık Celik, and Ibrahim Bozkurt. Industry 4.0 and smart production. *TEM Journal*, 10:799–805, 05 2021.

[11] Csaba Hegedus, Pál Varga, and Attila Frankó. Secure and trusted inter-cloud communications in the arrowhead framework. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pages 755–760, 2018.

[12] Jonny Herwan, Seisuke Kano, Ryabov Oleg, Hiroyuki Sawada, and Nagayoshi Kasashima. Cyber-physical system architecture for machining production line. In *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, pages 387–391, 2018.

[13] IEC/IEEE. Iec/ieee 60802 time-sensitive networking profile for industrial automation. https://www.ieee802.org/1/files/private/60802-drafts/d2/60802-d2-4.pdf, 2024. [Online; accessed 17-June-2024].

[14] Kyoung-Don Kang and Sang H. Son. Real-time data services for cyber physical systems. In *2008 The 28th International Conference on Distributed Computing Systems Workshops*, pages 483–488, 2008.

[15] Dániel Kozma, Pál Varga, and Kristóf Szabó. Achieving flexible digital production with the arrowhead workflow choreographer. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, pages 4503–4510, 2020.

[16] Vuk Lesi, Zivana Jakovljevic, and Miroslav Pajic. Synchronization of distributed controllers in cyber-physical systems. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 710–717, 2019.

[17] Lucia Lo Bello and Wilfried Steiner. A perspective on ieee time-sensitive networking for industrial communication and automation systems. *Proceedings of the IEEE*, 107(6):1094–1120, 2019.

[18] Jannis Ohms, Martin Böhm, and Diederich Wermser. Concept of a tsn to real-time wireless gateway in the context of 5g urllc. In *2020 8th International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 1–6, 2020.

[19] Peter A. Okeme, Anastasiia D. Skakun, and Alexander R. Muzalevskii. Transformation of factory to smart factory. In *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, pages 1499–1503, 2021.

[20] Miguel Saez, Francisco P. Maturana, Kira Barton, and Dawn M. Tilbury. Real-time manufacturing machine and system performance monitoring using internet of things. *IEEE Transactions on Automation Science and Engineering*, 15(4):1735–1748, 2018.

[21] Devarpita Sinha and Rajarshi Roy. Reviewing cyber-physical system as a part of smart factory in industry 4.0. *IEEE Engineering Management Review*, 48(2):103–117, 2020.

[22] Pal Varga, Fredrik Blomstedt, Luis Lino Ferreira, Jens Eliasson, Mats Johansson, Jerker Delsing, and Iker Martínez de Soria. Making system of systems interoperable – the core components of the arrowhead framework. *Journal of Network and Computer Applications*, 81:85–95, 2017.

ACKNOWLEDGMENT

The research leading to these results is funded by the EU Chips-JU organization under grant agreement 101112089, within the project AIMS5.0 and from the partners' national programs and funding authorities. Part of this work is created within the Arrowhead fPVN project, supported by the Chips-JU and its members, as well as by national funding authorities from involved countries under grant agreement no. 101111977.

Attila E. Frankó received his M.Sc. degree in 2019 in Electrical Engineering and his Ph.D. degree in 2024 from the Budapest University of Technology and Economics (BME), Hungary. He is currently working at the Department of Telecommunications and Artificial Intelligence (BME) as an assistant professor and also as a senior researcher at AITIA International Zrt. His research interests include data science, artificial intelligence, Industrial Internet of Things (IIOT), indoor positioning systems, anomaly detection in telecomunniation networks.

G. László Hollósi received his M.Sc. degree in electrical engineering with honors from the Budapest University of Technology and Economics (BME) in 2009, an engineering-economics degree from the Budapest Business School in 2014, and his Ph.D. degree from BME in 2025. From 2008 to 2009, he worked as a test automation engineer at Ericsson Hungary Kft. In 2009, he joined the Department of Telecommunications and Media Informatics (currently the Department of Telecommunications and Artificial Intelligence) at BME, where he currently serves as an assistant professor. Since 2026, he has held the position of chairman of the artificial intelligence section of the Scientific Association for Infocommunications. His research interests include data analysis, artificial intelligence, image processing, complex networks, indoor positioning, and time synchronization in packet-switched networks. He is a two-time recipient of the Pollák–Virág Award and serves as a regular reviewer for several journals, including IEEE Transactions on Measurement and Instrumentation, IEEE Transactions on Service and Network Management, IEEE Access, and Infocommunications Journal.

Dániel Ficzere is an Assistant Lecturer at the Budapest University of Technology and Economics (BME), Department of Telecommunications and Artificial Intelligence. He actively conducts research and publishes in the fields of telecommunications and artificial intelligence, specializing in generative AI for network operations, document processing, network science, and sports analytics. Alongside his academic work, he serves as the Secretary of the Artificial Intelligence Section of the Scientific Association for Infocommunications (HTE).

Pal Varga received his Ph.D. degree from the Budapest University of Technology and Economics, Hungary. He is currently an Associate Professor and head of Department of Telecommunications and Artificial Intelligence at Budapest University of Technology and Economics. His main research interests include communication systems, Cyber-Physical Systems and Industrial Internet of Things, network traffic analysis, end-to-end QoS and SLA issues – for which he is keen to apply hardware acceleration and artificial intelligence, machine learning techniques as well. Besides being a member of HTE, he is a senior member of IEEE, where he is active both in the IEEE ComSoc (Communication Society) and IEEE IES (Industrial Electronics Society) communities. He is Editorial Board member of the Sensors (MDPI) and Electronics (MDPI) journals, and the Editor-in-Chief of the Infocommunications Journal.

22nd International Conference on Network and Service Management

Evolving Network and Service Management: From Automation to Agentic Intelligence

Alcalá de Henares (Madrid), Spain // 26 - 30 October, 2026

The AnServApp Workshop

CALL FOR PAPERS

Important Dates

Paper Submission Deadline:
August 14, 2026

Notification of Acceptance:
September 8, 2026

Submission of Camera-ready Version:
September 14, 2026

Conference Date:
30 October 2026

Workshop Co-Chairs

Pal Varga
pvarga@tmit.bme.hu
Budapest University of Technology and Economics, Hungary

Tokunbo Makanju
makanju@cs.dal.ca
Dalhousie University, Canada

The International Workshop on Analytics for Service and Application Management (AnServApp) provides a forum for researchers and practitioners working on the use of data analytics, machine learning, artificial intelligence, and cognitive techniques for service and application management. The workshop welcomes contributions that address how operational data can be transformed into actionable insights, automated decisions, and trustworthy management processes.

Relevant approaches include predictive analytics, data mining, machine learning, deep learning, graph-based analytics, causal inference, anomaly detection, log and trace analysis, foundation models, large language models, and AI-assisted operations. The workshop is particularly interested in methods that support observability, performance management, fault detection and diagnosis, root-cause analysis, service assurance, intent-based management, resource optimization, security analytics, and closed-loop automation. AnServApp also encourages work that examines the reliability, explainability, robustness, and operational trustworthiness of analytics-driven management solutions.

The main goal of AnServApp is to present mature research results, emerging ideas, and work-in-progress contributions in the area of analytics and AI for service and application management. In addition to regular papers, short papers describing late-breaking advances, practical experiences, position papers, and reports from ongoing research are also welcome.

Topics of Interest

Topics of interest include but are not limited to:

- Analytics, machine learning, and AI for service and application management
- MLOps, LLMOps, observability, and automated operations
- Anomaly detection, incident prediction, and root-cause analysis
- Log, metric, event, and trace analytics
- Resource management & orchestration
- Multi-Agent Reinforcement Learning (RL)
- Distributed RL over communication networks
- Performance, reliability, and service assurance
- Resource management and optimization in cloud, edge, and hybrid environments
- Analytics for microservices, distributed applications, and cloud-native systems
- Intent-based, policy-based, and closed-loop service management
- Explainable, trustworthy, and robust AI for operational decision support
- Security analytics for service and application infrastructures
- Privacy-aware and federated analytics for operational data
- Foundation models and large language models for service management tasks
- Human-in-the-loop analytics and cognitive approaches to operations
- Practical experiences, datasets, benchmarks, and reproducibility studies in service and application analytics
- Verticals:
  - Lower carbon foot print applications / services / systems
  - Industry5.0 applications / services / systems
  - Smart cities and smart transportation services / systems
  - Social media apps / services / systems
  - Smart education services / systems
  - Edge, fog, cloud services / systems
  - Sustainability and resilience of applications / services / systems
  - Digital Twins for networks and services

Authors are invited to submit original unpublished papers not under review elsewhere. Papers should be submitted in **IEEE 2-column format**. Maximum paper length, including title, abstract, all figures, tables, and references, is **6 pages**.

Papers have to be submitted electronically in PDF format through the EDAS conference management system, accessible via EDAS: https://anservapp2026.edas.info

For accepted papers, at least one author is expected to register and present the paper in person at the workshop. Accepted papers will be published in the conference proceedings and submitted to IEEE Xplore.

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on

https://journals.ieeeauthorcenter.ieee.org/
Then click: "IEEE Author Tools for Journals"
- "Article Templates"
- "Templates for Transactions".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *Index Terms (Keywords)*. For the final version of accepted papers, please send the short cvs and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue
g) Document Object Identifier (DOI)

[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984. DOI: 10.1109/TEI.1984.298778

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to submit their papers electronically via the following portal address:

https://www.ojs.hte.hu/infocommunications_journal/about/submissions

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Editor-in-Chief: Pál Varga – pvarga@tmit.bme.hu

Associate Editor-in-Chief:

József Bíró – biro@tmit.bme.hu

László Bacsárdi – bacsardi@hit.bme.hu

IEEE International Conference on Communications

30 May – 4 June 2027 // Washington DC, USA

Confluence of Communications and AI

CALL FOR PAPERS

The 2027 IEEE International Conference on Communications (ICC 2027) will be held in Washington DC, USA from May 30 to June 3, 2027, with the theme of "Confluence of Communications and AI." This flagship conference of the IEEE Communications Society will feature a comprehensive high-quality technical program including 12 symposia, 13 tracks of Selected Areas in Communications and a variety of tutorials and workshops. IEEE ICC 2027 will also include an attractive industry program aimed at practitioners, with keynotes and panels from prominent research, industry and government leaders, business and industry panels, and technological exhibits. This conference will be fully in-person.

TOPICS OF INTEREST

Symposia

- Cognitive Radio & AI-Enabled Networks
- Communication & Information System Security
- Communication QoS, Reliability & Modelling
- Communications Software & Multimedia
- Communication Theory
- Green Communication Systems & Networks
- IoT & Sensor Networks
- Mobile & Wireless Networks
- Next-Generation Networking & Internet
- Optical Networks & Systems
- Signal Processing for Communications
- Wireless Communications

Selected Areas in Communications

- Aerial Communications
- Big Data
- Cloud/Edge Computing, Networking and Storage
- eHealth
- Integrated Sensing and Communications
- Machine Learning for Communications and Networking
- Molecular, Biological, and Multi-Scale Comm
- Next Generation Multiple Access
- Quantum Communications and Information Technology
- Reconfigurable Intelligent Surfaces
- Satellite and Space Communications
- Smart Grid Communications
- Social Networks

ORGANIZING COMMITTEE

General Chair
Bijan Jabbari, George Mason Univ., US

Executive Chair
Paul Cotae, Univ. of District of Columbia, US

TPC Chair
Hamid Jafarkhani, Univ. of California Irvine, US

TPC Vice Chairs
Ana Garcia Armada, Univ. Carlos III Madrid, ES
Shui Yu, Univ. of Technology Sydney, AU

Keynote Chair
Abbas Jamalipour, Univ. of Sydney, AU

IF&E Chairs
Nada Golmie, NIST, US
Hassan Kafi, T-Mobile, US

Workshop Program Co-chairs
Inkyu Lee, Korea Univ., KR
Wei Ni, CSIRO, AU
Hina Tabassum, York Univ., US

Tutorial Program Co-chairs
Zhi Tian, George Mason Univ., US
Falko Dressler, TU Berlin, DE

Awards Chair
Angel Lozano, Univ. Pompeu Fabra, ES

Travel Grants Co-chairs
Mojtaba Vaezi, Villanova Univ. US
Venkatesha Prasad, Delft Univ. of Tech., NL
Si-Hyeon Lee, KAIST, KR

Publications Co-chairs
Shahrokh Valaee, Univ. of Toronto, CA
Ruidong Li, Kanazawa Univ, JP

Publicity Chair
Petar Popovski, Aalborg Univ., DK
Liang Zhang, Univ. of Maryland Eastern Shore, US
Kohei Shiomoto, Tokyo City Univ., JP

Patronage Co-Chairs
Jerry Gibbon and Harry Sauberman, P.E., J & BG

Finance Chair
Robert Rassa, Raytheon, US

Operation Chair
Ashutosh Dutta, Johns Hopkins University, USA

GIMS Advisor
Dong-In Kim, Sungkyunkwan Univ., KR

GITC Advisor
Baek-Young Choi, MST, US

IMPORTANT DATES

Paper Submission: October 2, 2026
Acceptance Notification: January 15, 2027
Camera-Ready: February 19, 2027

Workshop Proposals: September 7, 2026
Tutorial Proposals: October 2, 2026
Panel Proposals: December 15, 2026

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS HUNGARY

FOUNDED IN 1949

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its 1000 individual members, the Scientific Association for Infocommunications (in Hungarian: Híradástechnikai és Informatikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **FERENC BANCSICS** • *elnok@hte.hu*

Secretary-General: **GÁBOR KOLLÁTH** • *kollath.gabor@hte.hu*

Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502

Phone: +36 1 353 1027

E-mail: *info@hte.hu*, Web: *www.hte.hu*