An Evolutionary Edge Computing Architecture for the Beyond 5G Era

Elli Kartsakli[1], Jordi Perez-Romero[2], Nikolaos Bartzoudis[3], Oriol Sallent[2], Oluwatayo Kolawole[4], Xin Tao[5], Swarup Kumar Mohalik[5], Tomasz Mach[4], Sige Liu[6], Yansha Deng[6], Gianluca Mandò[7], Angelos Antonopoulos[8], Valerio Frascolla[9], Semiha Kosu[10], Gökhan Kalem[10] Fred Buining[11], Eduardo Quiñones[1]

[1]Barcelona Supercomputing Center (BSC), Barcelona, Spain; [2]Signal Theory and Communications Dpt., Universitat Politècnica de Catalunya (UPC), Barcelona, Spain;[3]Telecommunications Technological Center of Catalonia (CTTC), Barcelona, Spain; [4]Samsung R&D Institute, Staines, United Kingdom; [5]Ericsson Research, Stockholm, Sweden; [6]Department of Engineering, King's College London (KCL), London, United Kingdom; Ground Transportation Systems Italia SRL, Sesto Fiorentino, Italy; [8]Nearby Computing S.L., Barcelona, Spain; [9]Intel, Neubiberg, Germany; [10]Turkcell Technology, Istanbul, Turkey; [11]HIRO-MicroDataCenters BV, Netherlands

***Abstract*— Beyond 5G (B5G) communication networks face the challenge of meeting the demanding requirements of various service types, including uRLLC, mIoT, eMBB, and emerging technologies like Extended Reality (XR). Edge computing can address these demands effectively because of the ability to bring computational power and resources closer to the source of data. Nevertheless, the realization of this potential necessitates an open, flexible, and automated architectural framework capable of supporting disaggregated applications and network designs. In this context, this paper introduces a novel architecture designed to advance the evolution of edge computing in B5G, developed within the EU-funded project VERGE. The proposed architecture is modular and scalable, guided by artificial intelligence (AI), and founded on three essential pillars: "edge for AI," "AI for edge," and "security, privacy, and trustworthiness for AI." After presenting this architecture, the paper showcases its applicability through the examination of two vertical use cases within the industrial and transportation domains.**

Keywords*— Edge computing; AI/ML-based optimization; security and trustworthiness; B5G/6G evolution; edge-cloud compute continuum; closed-loop automation

I. Introduction

Edge computing involves a connected ecosystem of highly heterogeneous computing elements, distributed among end-devices, access and core network, and sharing boundaries with central cloud infrastructures [1], [2]. There is an ongoing standardization effort to enable edge computing in beyond 5G (B5G) networks, mainly driven by 3GPP (refer to Rel-18 [3] and new features in Rel-19), and by the European Telecommunications Standards Institute (ETSI) (refer to the Multi-access Edge Computing (MEC) specification [4]).

Edge computing can enable several key innovations in B5G, such as dynamic network slicing, flexible functional splits, improved network determinism, and adaptive Virtual Network Function (VNF) placement and scaling. Furthermore, the introduction of artificial intelligence (AI) and machine learning (ML) in resource orchestration will enable a dual-layer control at the edge. The first layer concerns a new level of closed-loop programmability and automation, especially when near real-time decisions need to be made while handling massive amounts of data close to the end users. The second layer serves digital sovereignty, created by means of a powerful local distributed edge infrastructure that provides services to manage identities, applications, and dataspaces for multiple tenants.

In addition to empowering B5G network optimization and automation, edge computing is a key vertical service enabler across multiple sectors [5]. Real-time immersive applications, as the ones based on *eXtended Reality (XR)* and holographic representations, are progressively emerging, enabling innovative services like online gaming, robotic teleoperation and remote education. On the other hand, big data analytic pipelines, processing the massive amount of data generated by distributed *Internet of Things (IoT)* deployments, are leveraged in a wide range of applications (e.g., digital twins [6]) for smart cities, Industry 4.0, autonomous vehicles, etc.

Such applications pose significant and diverse challenges on existing network and computing infrastructures. Current edge-enabled 5G architectures lack the required level of flexibility, openness and automation, and the mechanisms to support distributed and disaggregated application and network designs that are needed by such next generation services. Besides, even though edge computing has been widely considered within 5G networks, the adopted approaches have been mainly driven by specific use case requirements leading to a fragmented architectural landscape with respect to the edge deployments and performance aspects [7][8]. Hence, further evolution and closer synergy between the B5G and the edge computing paradigms are needed to ensure the real-time responsiveness and fast computation capacity needed to ensure enhanced and dynamic user experience [9].

To address these gaps and to fully exploit the potential of edge computing, this paper proposes an evolved edge computing architecture integrated with the B5G network fabric. This architecture is developed within the EU-funded research project VERGE [10]. The proposed design aims to enable the seamless execution of cloud-native services, including disaggregated Radio Access Network (RAN) and core network functions, distributed AI, and big data workflows, while leveraging data-driven, AI/ML-based solutions for edge and network optimization. Simultaneously, it ensures that the AI-based solutions themselves are secure and trustworthy. The proposed architecture is modular and scalable, powered through secure data-driven and AI-based solutions that enable its adaptability to the requirements of B5G and forthcoming 6G applications.

The architecture presented in this paper (henceforth referred to as the "VERGE architecture") is built around three main pillars: i) "Edge for AI" (hereafter referred to as Edge4AI), a flexible, modular and converged edge platform design that unifies the lifecycle management (LCM) and closed-loop automation for cloud-native applications and network services across a unified edge-cloud compute continuum; ii) "AI for Edge" (AI4Edge), an AI-powered portfolio of solutions that leverages the multitude of metrics provided by the monitoring mechanisms to manage and orchestrate the computing and network resources; and iii) "Security, Privacy and Trustworthiness for AI" (SPT4AI), a suite of methods and tools to ensure data and AI-based model privacy, security of the AI-based models against adversarial attacks, their safe training and execution, and their explainability for different stakeholders.

The structure of the paper is as follows. After introducing the novel architecture in Section II, details of the key pillars are provided in Sections III, IV and V. Section VI illustrates the applicability of the architecture in two vertical use case examples, for XR-enabled services in an industrial environment and IoT-driven autonomous tram services in a smart city. Finally, Section VI concludes the paper.

II. Proposed Edge Computing Architecture

A high-level view of the proposed VERGE architecture is shown in Fig. 1. A highly heterogeneous infrastructure is depicted at the bottom, consisting of diverse edge computing resources (from the Far Edge to the Near Edge and the Cloud) embedded in the end-to-end (E2E) B5G network. Different types of users are considered, connected through heterogeneous RAN deployments (disaggregated, relay-enabled, etc.) and leveraging MEC services. VERGE architecture is shown in the upper part of Fig. 1, featuring the three aforementioned pillars, further discussed next.

The *Edge4AI* layer forms an AI-powered platform to facilitate the deployment and execution of cloud-native services and network functions (coming from the *Application layer*) over the heterogeneous pool of connected edge and cloud resources. The Edge4AI *virtualization layer* provides a unified view of the communication and computational resources, forming an edge-cloud compute continuum that is tightly integrated with the B5G communication fabric. To fully leverage such resources, Edge4AI contains: i) the *Orchestration, Management and Control layer*, handling the orchestration of services and infrastructures, and the control of the RAN elements; ii) the *Cognitive Framework*, enabling the LCM of the developed AI/ML solutions; iii) the *Distributed Knowledge Base (DKB)* where all generated knowledge (e.g., trained AI/ML models, datasets, metadata) are registered; and iv) the *Data Access* layer, responsible for collecting all the relevant observability data across the entire deployed system. The generated datasets from the data access layer, as well as any additional external or synthetic datasets employed for the training of AI/ML models, are stored in an *Open Dataspace*, enabling their reutilization and transparent usage.

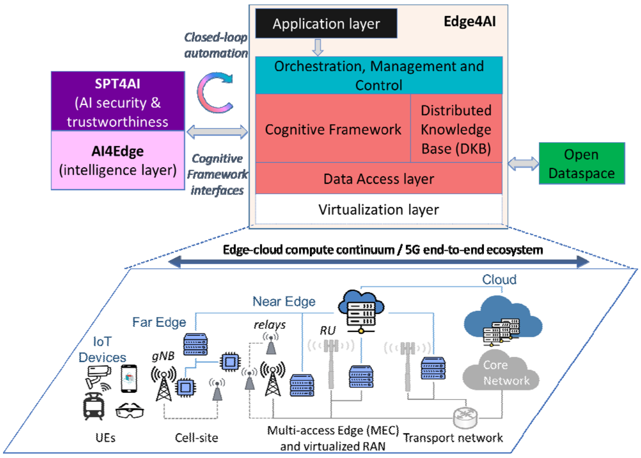

Closed-loop automation
Application layer
Edge4AI
Orchestration, Management and Control
SPT4AI (AI security & trustworthiness
AI4Edge (intelligence layer)
Cognitive Framework
Distributed Knowledge Base (DKB)
Cognitive Framework interfaces
Data Access layer
Open Dataspace
Virtualization layer
Edge-cloud compute continuum / 5G end-to-end ecosystem
Cloud
Near Edge
Far Edge
IoT Devices
relays
RU
gNB
Core Network
UEs
Cell-site
Multi-access Edge (MEC) and virtualized RAN
Transport network

Fig. 1. The key building blocks of the proposed edge computing architecture

The *AI4Edge* forms the intelligence layer encompassing all the AI/ML models that can be used for the automated management and optimization of communication and computing resources, as well as for supporting advanced data-driven applications able to interact with their environment and provide immersive services to the users. The AI4Edge layer, facilitated by the Edge4AI cognitive framework functionalities and interfaces, specifies the model-specific methods for: i) AI/ML training and validation; ii) AI/ML model monitoring and management (e.g., retraining if needed); and iii) AI/ML model inference, in which trained models are used to optimize Edge4AI components (e.g., orchestrators, RAN controllers, etc.) The training of these models can make use of datasets included in the *Open Dataspace*.

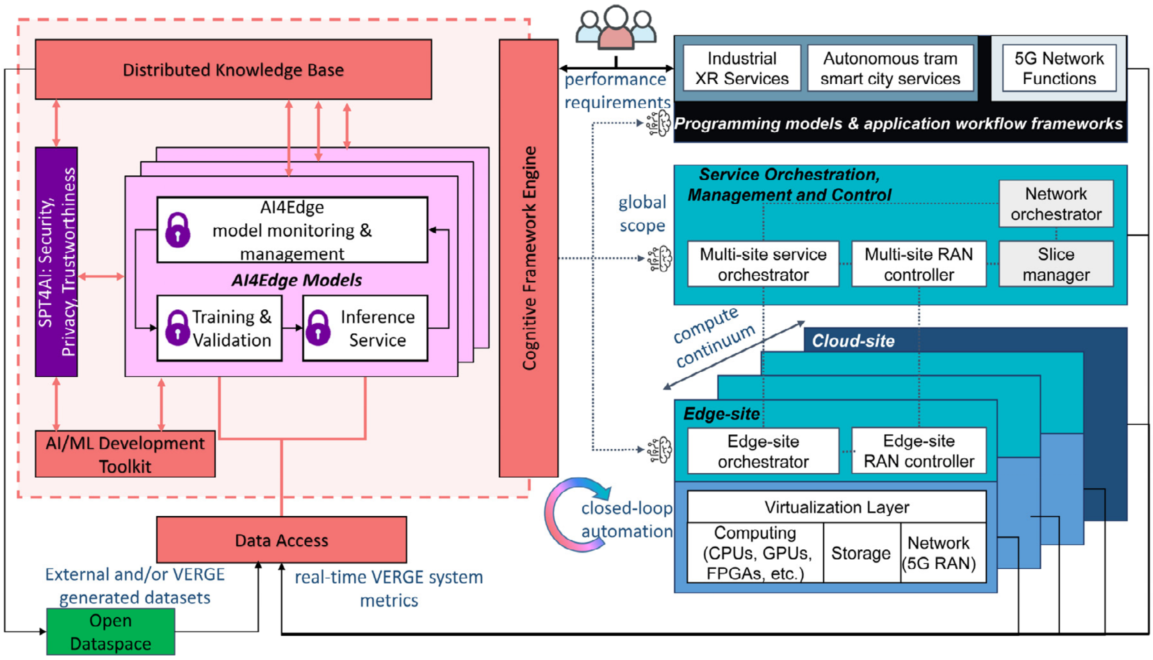
Finally, the *SPT4AI* layer defines a set of security, privacy and trustworthiness processes applied to the AI4Edge models. The interoperability between the diverse portfolio of security/privacy/safety/explainability solutions of SPT4AI with the AI4Edge platform is ensured through the open interfaces provided by the cognitive framework. These interfaces provide a common way to handle the complete lifecycle (including training, validation, deployment, inference, and monitoring) of the AI/ML models that are deployed by the Orchestration, Management and Control elements to enforce the intelligent decisions.

A more detailed view of the proposed architecture is given in Fig. 2. The architectural components of each pillar depicted in the figure are described in the following sections, namely the Edge4AI (Section III), the AI4Edge (Section IV) and the SPT4AI (Section V), stressing their key capabilities and interactions.

III. The Edge4AI Layer

A. Programming Models and Application Workflow Frameworks

A cloud-native design approach is adopted, where both application and network functions, composing the



Distributed Knowledge Base
SPT4AI: Security, Privacy, Trustworthiness
AI4Edge model monitoring & management
AI4Edge Models
Training & Validation
Inference Service
AI/ML Development Toolkit
Cognitive Framework Engine
Data Access
External and/or VERGE generated datasets
Open Dataspace
real-time VERGE system metrics
performance requirements
global scope
closed-loop automation
compute continuum
Industrial XR Services
Autonomous tram smart city services
5G Network Functions
Programming models & application workflow frameworks
Service Orchestration, Management and Control
Network orchestrator
Multi-site service orchestrator
Multi-site RAN controller
Slice manager
Cloud-site
Edge-site
Edge-site orchestrator
Edge-site RAN controller
Virtualization Layer
Computing (CPUs, GPUs, FPGAs, etc.)
Storage
Network (5G RAN)

Fig. 2. A novel architecture for the edge computing evolution

Application layer of Fig. 1, are primarily packaged as containers. However, to ensure backwards compatibility, other implementation technologies such as Virtual Machines (VMs) or physical functions are supported. The functions/services running at the edge include:

- *Application functions* implementing the different use cases that can exploit the proposed architecture.
- *5G/B5G network functions* for the RAN, the core and the control plane running at the edge. Specific examples include: i) virtualized RAN functions, e.g., implementing Centralized and Distributed Unit functionalities (CU/DU), aligned with the trend for disaggregated RAN functions promoted by 3GPP [11] and the Open RAN (O-RAN) Alliance [12]; ii) virtualized core functions, e.g., User Plane Function (UPF), Network Exposure Function (NEF), etc.; and iii) virtualized RAN control elements.
- *AI/ML functions* implementing the AI4Edge intelligence layer, including the training and inference of AI models that may run at the edge, as well as distributed learning methods.

Leveraging the flexibility offered by the cloud-native application design, this evolved edge architecture supports different levels of distribution mechanisms for splitting the computation across the available edge and cloud resources: i) *in-node splitting* of computing tasks, i.e., within the same multi-accelerator platform; ii) *horizontal distribution* of computation among peer edge nodes; and iii) *vertical distribution*, between end users, edge and cloud.

This key innovation of splitting the computation, e.g., splitting the processing tasks associated to different layers of a Deep Neural Network (DNN), is provided by employing programming models and application workflow frameworks supported by the Edge4AI layer. The decision of selecting the optimal split point to offload computation tasks across the available edge and cloud resources is made based on the knowledge acquired from the AI4Edge layer. Thus, the computational burden can be alleviated by taking into consideration the available resources of the devices and edge servers, application latency requirements, and the resource constraints of a given use case deployment.

Moreover, the most suitable programming models and practices from the embedded, High Performance Computing (HPC) and AI domains are employed and adapted for the implementation of highly efficient application workflows at the edge-cloud compute continuum. A key innovation towards this direction is the design of an adaptive virtualization layer specifically targeting programmable and accelerated hardware platforms, enabling the dynamic reconfiguration of functions across embedded AI accelerators and general-purpose computing elements.

B. Service Orchestration, Management and Control

The high flexibility in the deployment of cloud-native functions over a highly heterogeneous edge-to-cloud compute continuum infrastructure and the support for multi-tenancy and distributed execution call for a unified design for the service orchestration, management and control planes. A *distributed/ hierarchical approach* is adopted for this layer, considering both a *global multi-site view* and a *local edge-site perspective*, to better deal with the distributed nature of the underlying infrastructure. On the one hand, this design enables the E2E optimization across multiple sites, facilitating functionalities such as service migration, edge federation, mobility support, etc. On the other hand, having a local orchestration layer at each edge-site enhances the flexibility and modularity of the system, enabling localized optimization actions with reduced management overhead (e.g., intra-node orchestration or single-site interference management). The key functionalities of the Service Orchestration, Management and Control layer are:

- *Service orchestration*, handling the service onboarding and LCM of cloud-native applications and AI/ML workflows, and the orchestration of the underlying computing resources at both global and local scale.

- *RAN control*, responsible for the management of the RAN elements, consisting of i) a multi-site RAN controller for optimization across multiple RAN sites and ii) an edge-site component for localized actions, often at a near real-time scale. When implemented as containers, the RAN controllers are also managed by the service orchestrator.

Two additional network-related functionalities are also supported, only considering their scope within the edge-cloud continuum domain:

- *Network orchestration*, handled by the Network Function Virtualization Orchestrator (NFVO) in the ETSI NFV specifications. The NFVO is responsible for the LCM of network services composed by multiple VNFs running at the edge (whereas an E2E NFVO may reside beyond the considered edge-cloud continuum).
- *Slice management*, responsible for the network slice provisioning, i.e., the slice instantiation, operation, modification, and termination.

C. Data Access and Cognitive Framework

The *data access laye*r and *cognitive framework* of the Edge4AI are the key enablers of closed-loop automation in the proposed architecture. The cognitive framework fuels the AI4Edge layer with the necessary data, collected by the data access layer, and supports the lifecycle of the AI4Edge AI/ML models. In turn, the output of these models provides intelligent decisions to the Edge4AI service management, orchestration and control layer, optimizing and automating multiple aspects of the underlying system.

The *data access* layer is responsible for the extraction of relevant metrics from all layers of the architecture, including infrastructure, platform and services. To that end, the data access layer provides all the necessary distributed agents for the ingestion of RAN and core metrics, exposed by the B5G network, edge platform telemetry (e.g., Central Processing Unit (CPU)/storage/memory utilization, etc.), and application-related requirements, exposed by the employed programming models. The data access layer may also collect data from external sources, e.g., synthetic datasets, datasets created over extended operation of the deployed system, etc., made available through the Open Dataspace environment. The data access layer is responsible for integrating and analyzing the collected data (exploratory data analysis), creating datasets to be used by the AI4Edge models (stored in the data registry) and providing their access to the respective AI models.

The *cognitive framework* provides a common framework and interfaces for the development, deployment and LCM of the AI4Edge AI/ML models, aligned with the best practices defined in ML Operations (MLOps) [14]. The framework includes:

- *Cognitive framework engine*, providing all needed functionalities and Application Programming Interfaces (APIs) to enable model training, monitoring and inference serving across the entire architecture, and to ensure the exposure of all relevant services to the Service Orchestration, Management and Control layer. Additional APIs enable end users to introduce specific performance requirements (e.g., on AI/ML safety), as well as enable the ingestion of the right level of data by the AI4Edge layer, as needed by each AI/ML model.
- *AI/ML development toolkit*, containing software tools and libraries that are required by the AI4Edge and SPT4AI pillars, for the design, validation and trustworthiness of AI/ML models.
- *Distributed knowledge base (DKB)*, containing all relevant data and knowledge generated and required by the AI4Edge AI/ML models, such as the generated datasets, trained models, and other relevant metadata and platform metrics. The DKB is deployed in each edge-site, storing only locally relevant metadata and any other information needed by the AI4Edge models running at each location.

IV. The AI4Edge Layer

The AI4Edge layer represents a comprehensive suite of AI/ML models designed to (1) manage the entire lifecycle of computing, communication, and networking resources, and (2) power the network edge applications in an efficient and scalable manner, leveraging the Edge4AI capabilities.

The key model-specific functionalities provided by each AI/ML model within the AI4Edge layer include:

- *Training & Validation*: This functionality encompasses the full training pipeline of AI/ML models. It includes all relevant data management procedures for preparing the data collected by the data access layer and storing the trained model and necessary metadata in the DKB.
- *Inference Service*: This refers to the application of the trained model in empowering intelligent decision-making processes within the Edge4AI orchestration, management, and control entities (for example, the local or multi-site orchestrators, the RAN controllers, etc.). Various implementations for serving inference services can be considered, whether through APIs or packaged as an AI/ML-enabled application.
- *AI4Edge Model Monitoring and Management*: This functionality is in charge of (1) monitoring the quality of the inference predictions based on user-defined Key Performance Indicators (KPIs), prompting the retraining of the model if necessary, and (2) managing the model serving and deployment processes.

These AI/ML models, though inherently unique, are unified under the cognitive framework, enabling them to interact seamlessly with the underlying system. However, it should be noted that, depending on the scope of each AI/ML model, some functionalities (e.g., the training of a complex model) might not be running at the edge.

The proposed architecture also takes some initial steps towards resolving the challenges associated with the widespread adoption of multiple, independent AI/ML solutions. Such mechanisms detect and resolve potential conflicts between the independent AI/ML-based decisions, ensuring that the system behaviour is stable and efficient.

V. The SPT4AI Layer

The SPT4AI layer provides a set of methodologies and tools for *secure*, *private*, *safe* and *explainable* operations of the AI4Edge models, thereby increasing their *trustworthiness*. Various functionalities of the SPT4AI layer are exposed through the cognitive framework APIs and are invoked during the design and deployment phases of the AI4Edge models. These are summarized as follows:

- *Security and privacy*: providing a methodology to protect the AI4Edge models against potential adversarial attacks and functional failures, SPT4AI implements targeted threat analysis methods and recommends efficient and robust mitigation methods. It also provides methods to ensure the security and privacy of the sensitive information carried by the data and models against attacks.
- *Design time verification*: providing formal and quantitative verification results of safety metrics for AI4Edge models, specifically targeting Reinforcement Learning (RL) models. It collects safety requirements from stakeholders (e.g., service consumers, model developers), builds and maintains environment models from training data, and integrates off-the-shelf open-source tools to produce reports about the safety KPI satisfaction of the ML models. The reports, containing possible counterexamples and insights, can be used to improve the models through, for example, additional targeted training data. These reports can also be used to explain the compliance of the model to the safety KPIs and identify unsafe operating regions.
- *Run-time verification*: monitoring the deployed ML models and the application environment for potential violation of the safety KPIs (e.g., range violation, wrong sequence of actions) during the system run-time. The reports can guide the cognitive framework to decide fallback procedures and/or model re-training.
- *Explainability and interpretability*: Model explainability methods include both interpretability (i.e., the ability to present the cause-and-effect relationship between the model input and output), and the logic through which the relationship is established, e.g., feature attribution explaining to what extent a decision is impacted by an input feature. Explainability contributes to accountability, reliability and transparency. It is achieved either via the adoption of *post-hoc explainability techniques* or via the design of *inherently interpretable built-in models*. The incorporation of such methodologies fosters trustworthiness in AI4Edge solutions, as well as their legal and regulatory compliance.

VI. B5G Edge-Enabled Use Case Examples

In order to show the suitability and versatility of the VERGE architecture across vertical use cases in B5G-enabled networks, two examples from the industrial and transportation domains are elaborated: *XR-driven edge-enabled industrial B5G applications* and *autonomous tram services for safety and entertainment in a smart city environment*.

A. XR-driven edge-enabled industrial B5G applications

The Fourth Industrial Revolution, also known as Industry 4.0, represents the digital innovation effort across several industrial sectors. Industry 4.0 leverages cutting-edge technologies, such as AI, IoT, and automation, to establish intelligent factories that increases productivity, reduces cost and facilitates real-time monitoring and decision-making. XR is a key technology that can contribute to productivity and monitoring via immersive services such as *robot teleoperation* and *cooperative product design* [15].

Consider, for instance, an information-rich industrial environment in which human operators must seamlessly interact and control robotic vehicles remotely for security reasons. Typically, visual contact or video feedback is used to determine the course of the vehicles, often resulting in disorientation and reduced piloting efficiency. The integration of XR technology, blending virtual and physical elements and overlaying relevant information, can significantly enhance the operators' experience, immersing them in the robots' coordinates system for more intuitive control. Edge computing can play a key role in such scenarios, bringing the processing of the massive amount of data (environment structure, maps, etc.) closer to both robots and end users, thus reducing latency and ensuring immediate responses. Furthermore, keeping the data at the edge, and especially in private edge servers in industrial facilities, mitigates security and privacy concerns. With respect to communications, ultra-low latency connection with mobility support is required, with sufficient coverage, bandwidth and reliability to serve potentially massive deployments of robots, pushing conventional access technology to its utmost.

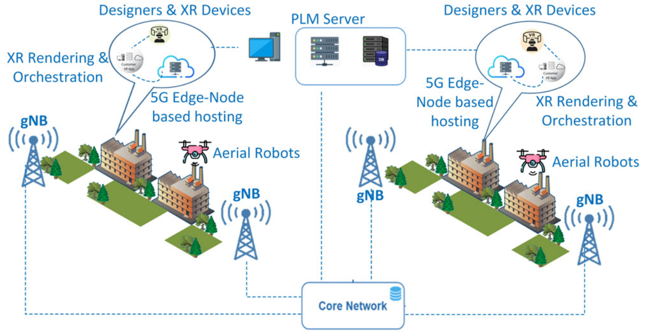

Designers & XR Devices
PLM Server
Designers & XR Devices
XR Rendering & Orchestration
5G Edge-Node based hosting
gNB
Aerial Robots
gNB
5G Edge-Node based hosting
XR Rendering & Orchestration
gNB
Aerial Robots
gNB
Core Network

Fig. 3. XR-driven edge-enabled industrial B5G use case scenario

XR can also drastically change and enhance the industrial product design process, which is currently impacted by the high latencies in processing and visualizing large volumes of 3D Computer Aided Design (CAD) files. Using XR tools integrated with the industrial Product Lifecycle Management (PLM) software, designers from remote locations can work collaboratively and simultaneously on the same product design. This enables the efficient and faster transfer of knowledge, with an immense reduction on design time and travel cost. Currently, XR data is typically stored in the cloud, but computation continues to occur on the mobile end device, with limited capabilities. This limitation can be overcome by hosting the XR application at the edge and only stream the content to the XR device/user. Such a solution requires fast and secure exchange of big data volumes between locations, as well as a low-latency connectivity between the PLM and XR platforms, stressing the need for B5G solutions, especially as the number of concurrent design sessions increases.

Fig. 3 illustrates two XR-aided industrial applications for remote robot handling and collaborative product design, enabled by the synergy of edge computing with B5G communications. The key involved stakeholders include: i) the *edge infrastructure provider*, who develops and maintains the physical and virtual infrastructure with the computational resources required to support the XR services; ii) the *mobile network operator (MNO)*, who provides wireless access, broadband network and services that allow designers and human operators of the robots to stay connected; and iii) the *end users*, including the designers who actively participate in the collaborative design process, and the robot manipulators, controlling remotely the robots that transmit real-time video and sensor data to the edge; and iv) the *application provider*, who develops and maintains the XR rendering application.

On this basis, the main capabilities that the proposed architecture offers to support this use case are the following:

- At the *Edge4AI* layer, the offloading of the heavy computational XR processes to the edge is supported, considering the device-specific limitations for rendering the huge volume of data (e.g., CAD files can be composed of several million polygons, whereas standalone XR glasses can only effectively visualize up to one million polygons). This is achieved, on the one hand, through the development of cloud-native middleware solutions at the application layer for the XR rendering and streaming. On the other hand, advanced *edge-site* and *multi-site orchestration* supports the flexible deployment, scaling and overall LCM of the XR software components, especially as the number of end users (designers or robot operators) increases.
- The AI4Edge layer supports *predictive intelligent analytics* to forecast the future resource needs and allocate network slices proactively, thus dealing with the stringent latency and bandwidth requirements of the XR services. Furthermore, *AI-driven resource management and network function configuration* based on real-time demands is supported, leveraging the system's capability to collect runtime metrics. To ensure the coherent integration between the AI components *multi-level multi-agent mechanisms* will be investigated, preventing potential conflicts in the decision-making process.
- Security and privacy are fundamental concerns in industrial environments and are considered in multiple levels across the proposed architecture and the *SPT4AI* layer. First, to provide inherent security support, Non-Public Networks (NPNs) deployments are considered, through either i) standalone NPNs totally separated from the public network, or ii) public networks integrated with NPNs, in which the NPN is provided in the form of a network slice of the MNO. To maximize privacy, scalability and efficiency, *distributed learning methods* are employed, allowing local training without centralized data sharing to dynamically allocate computation resources across the compute continuum.. Finally, *smart defense techniques* is provided to overcome the effects of malicious attacks against AI/ML models which target specific vulnerabilities due to the distributed and heterogeneous nature of edge deployments.

B. Autonomous tram services for safety and entertainment in a smart city environment

The concept of autonomous driving in Light Rail Transit (LRT) tram public transport systems heavily relies on multiple sensors, such as cameras, radars, light detection and ranging (LIDAR) systems, etc. The sensor data coupled with Bayesian and AI processes can provide the necessary level of perception and actuation in real-time. However, the implementation of such autonomous systems poses challenging requirements. Since these services are exposed to several critical uncontrollable events such as pedestrians, vehicles, and obstacles, a much higher level of situational awareness and more dynamic interaction must be supported. Such capabilities can be offered by edge computing and B5G communication technologies, supporting the orchestration of tram autonomy functions and dedicated network slices for safety-related critical and non-critical data. These mechanisms can provide the necessary ultra-reliable low-latency connectivity, high computational capability closer to the data sources, mobility support and dynamic reconfigurability needed to implement different services of the autonomous tram of the future.

At the same time, smart cities are adopting sensing, computing and communication technologies to provide innovative services for a more efficient, safe and sustainable city management and enhanced quality of life. Fueled by the wide IoT penetration, cities are collecting massive volumes of data, which, through AI and big data technologies, is transformed into valuable and actionable knowledge, able to automate and optimize several city aspects. In such highly distributed and heterogeneous environments, edge computing can be leveraged to fuse together information coming from both the sensor-equipped trams and the city to detect and anticipate hazards that may lie along the trajectory of the tram, beyond the visibility of the tram sensors. Furthermore, innovative XR entertainment services with demanding latency and processing requirements can be supported, e.g., providing immersive touristic information for people moving by tram or other means within specific areas of interest.

Fig. 4 illustrates the components of this use case, which involves applications (e.g., track occupation monitoring and obstacle/hazard detection) aimed at improving the safety and operational efficiency of the autonomous tram in a smart city environment, and at providing immersive entertainment services for passengers. The key stakeholders include: i) *tram operators*, owning the autonomous tram infrastructure (e.g., railway lines, interlocking systems, operational control centers, etc.); ii) *edge infrastructure* and *IoT providers*, offering the computational resources at the edge and the smart city infrastructure (e.g., smart sensors, cameras, etc.); iii) *MNO*, offering wireless connectivity between the autonomous tram and the infrastructure; iv) *AI models/ application developers*, supporting services like object detection, classification and tracking, and entertainment services targeting the tram passengers; and v) *end users*, consuming the safety and entertainment services.

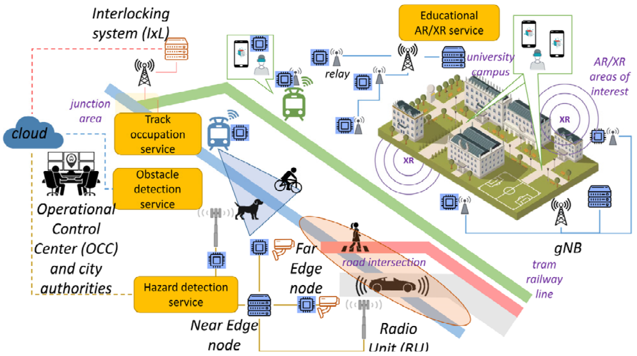


Fig. 4. Autonomous tram services for safety and entertainment in a smart city environment use case scenario

On this basis, the main capabilities that the proposed architecture offers to support this use case are the following:

- The *Edge4AI* layer supports the decomposition of monolithic services into microservices, enabling their flexible and scalable deployment, which is crucial given the heterogeneous and distributed nature of the smart city IoT devices. Furthermore, the adoption of HPC and distributed programming models at the Edge4AI layer breaks complex data analytics pipelines into linked tasks, enabling their parallel execution.
- Exploiting the Edge4AI features, the availability of intelligent task-based scheduling algorithms can allocate tasks to the most suitable computing resources, taking into account task-specific requirements in terms of latency, computational needs, data locality, etc. AI4Edge also supports the optimal splitting of complex tasks (e.g., 3D model reconstruction or AI-based workflows for video processing and object detection), and their placement at the most appropriate resources across the compute continuum, balancing latency requirements, energy consumption constraints and resource utilization.
- Ensuring the robustness and trustworthiness of the employed AI algorithms for the autonomous tram operation is essential. Insufficient distribution of training data or lesser generalizability of the trained model can lead to unsafe model behavior in new and unseen scenarios. Moreover, malicious attacks can substantially compromise the security of these AI-driven systems, often with disastrous results [16]. The SPT4AI layer offers the mechanisms to detect and mitigate security risks. It also ensures safety of the decisions output by the AI/ML models. In addition, by obtaining uncertainty information from AI/ML models (e.g., DNN predictions), SPT4AI can increase reliability and reduce the risk of making wrong decisions.

VII. Conclusions

This paper proposes a modular and scalable architecture to support edge computing evolution demands towards B5G, introduced by VERGE [10]. The proposed architecture opens the door for novel solutions to enable AI-enabled automation and intelligent decision-making across an integrated edge-cloud computing ecosystem, while ensuring security, privacy, and trustworthiness of the employed AI models. Two use case scenarios leveraging XR and IoT technologies also illustrate the potential benefits of the proposed architecture in the industrial and transportation vertical sectors.

Acknowledgement

Project VERGE has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101096034. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them. UK participants in VERGE are supported by UKRI grant numbers 10071211 (Samsung Electronics (UK) Limited) and 10061781 (King's College London).

The authors would like to thank the whole VERGE consortium for its contributions to shape the concepts, research challenges, architectural solution and use cases.

References

[1] A. Yousefpour et al., "All one needs to know about fog computing and related edge computing paradigms: A complete survey", Journal of Systems Architecture, vol. 98, September 2019.

[2] B. Gu et al., "Context-Aware Task Offloading for Multi-Access Edge Computing: Matching with Externalities," IEEE Global Communications Conference (GLOBECOM), UAE, December 2018.

[3] 3GPP TS 23.501 v18.2.1, "System architecture for the 5G System (5GS); Stage 2", June, 2023.

[4] ETSI GS MEC 003, v3.1.1, "Multi-access Edge Computing (MEC); Framework and Reference Architecture", March 2022.

[5] W. Saad, M. Bennis, M. Chen, "A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems", IEEE Network, vol.34, 2020.

[6] T. Do-Duy, D. Van Huynh, O. A. Dobre, B. Canberk and T. Q. Duong, "Digital Twin-Aided Intelligent Offloading With Edge Selection in Mobile Edge Computing," in IEEE Wireless Communications Letters, vol. 11, no. 4, pp. 806-810, April 2022.

[7] 5G-PPP Architecture Working Group, "View on 5G Architecture v5.0", White paper, Oct. 2021, doi: 10.5281/zenodo.5155657.

[8] 5G-PPP Technology Board Working Group, 5G-IA's Trials Working Group, "Edge Computing for 5G Networks v1.0", White paper, Jan. 2021, doi: 10.5281/zenodo.3698117.

[9] AIOTI, "High Priority Edge Computing Standardisation Gaps and Relevant SDOs", April, 2022.

[10] https://www.verge-project.eu/

[11] 3GPP TS 38.401 v17.5.0, "NG-RAN; Architecture description", June, 2023.

[12] A. Akman, et al, "O-RAN Minimum Viable Plan and Acceleration towards Commercialization", O-RAN Alliance White Paper, June 2021.

[13] E. Kartsakli, et al., "AI-powered edge computing evolution for beyond 5G communication networks", 2023 EuCNC/6G Summit, Gothenburg Sweden, June 2023.

[14] K. Salama, J- Kazmierczak, D. Schut, "Practitioners guide to MLOps: A framework for continuous delivery and automation of machine learning", Google Cloud, White paper, May 2021.

[15] T. Taleb *et al.*, "Toward Supporting XR Services: Architecture and Enablers," *IEEE Internet Things J.*, vol. 10, no. 4, Feb., 2023.

[16] C. Sitawarin, A.N. Bhagoji, A. Mosenia, M. Chiang, P. Mittal,, "Darts: Deceiving autonomous cars with toxic signs", arXiv, 2018, [Online]: https://doi.org/10.48550/arXiv.1802.06430.