6G WHITE PAPER ON EDGE INTELLIGENCE

6G Research Visions, No. 8
June 2020

6G FLAGSHIP
UNIVERSITY OF OULU

Table of Contents

6G White Paper on Edge Intelligence

6G Research Visions, No. 8
ISSN 2669-9621 (print)
ISSN 2669-963X (online)
ISBN 978-952-62-2677-4 (online)

Authors
Ella Peltonen, University of Oulu, Finland, ella.peltonen@oulu.fi · Mehdi Bennis, University of Oulu, Finland, mehdi.bennis@oulu.fi · Michele Capobianco, Capobianco, Italy, michele@capobianco.net · Merouane Debbah, Huawei, France, merouane.debbah@huawei.com · Aaron Ding, TU Delft, Netherlands, aaron.ding@tudelft.nl · Felipe Gil-Castiñeira, University of Vigo, Spain, xil@gti.uvigo.es · Marko Jurmu, VTT Technical Research Centre of Finland, Finland, marko.jurmu@vtt.fi · Teemu Karvonen, University of Oulu, Finland, teemu.3.karvonen@oulu.fi · Markus Kelanti, University of Oulu, Finland, markus.kelanti@oulu.fi · Adrian Kliks, Poznan University of Technology, Poland, adrian.kliks@put.poznan.pl · Teemu Leppänen, University of Oulu, Finland, teemu.leppanen@oulu.fi · Lauri Lovén, University of Oulu, Finland, lauri.loven@oulu.fi · Tommi Mikkonen, University of Helsinki, Finland, tommi.mikkonen@helsinki.fi · Ashwin Rao, University of Helsinki, Finland, ashwin.rao@helsinki.fi · Sumudu Samarakoon, University of Oulu, Finland, sumudu.samarakoon@oulu.fi · Kari Seppänen, VTT Technical Research Centre of Finland, Finland, kari.seppanen@vtt.fi · Paweł Sroka, Poznan University of Technology, Poland, pawel.sroka@put.poznan.pl · Sasu Tarkoma, University of Helsinki, Finland, sasu.tarkoma@helsinki.fi · Tingting Yang, Pengcheng Laboratory, China, yangtt@pcl.ac.cn

Please cite:
Peltonen, E., Bennis, M., Capobianco, M., Debbah, M., Ding, A., Gil-Castiñeira, F., Jurmu, M., Karvonen, T., Kelanti, M., Kliks, A., Leppänen, T., Lovén, L., Mikkonen, T., Rao, A., Samarakoon, S., Seppänen, K., Sroka, P., Tarkoma, S., & Yang, T. (2020). *6G White Paper on Edge Intelligence* [White paper]. (6G Research Visions, No. 8). University of Oulu. ***http://urn.fi/urn:isbn:9789526226774***

6G Flagship, University of Oulu, Finland
June 2020

Acknowledgement

This white paper has been written by an international expert group, led by the Finnish 6G Flagship program (6gflagship.com) at the University of Oulu, within a series of twelve 6G white papers.

Abstract

In this white paper, we provide a vision for 6G edge intelligence. Moving toward 5G and beyond future 6G networks, intelligent solutions utilizing data-driven machine learning and artificial intelligence will become crucial for several real-world applications, including but not limited to more efficient manufacturing, novel personal smart device environments and experiences, urban computing, and autonomous traffic settings. We sent edge computing with other 6G enablers as a key component to establish the future 2030 intelligent Internet technologies shown in this series of 6G white papers.

In this white paper, we focus on the domains of edge-computing infrastructure and platforms, data and edge network management, software development for edge, and real-time and distributed training of ML/AI algorithms, as well as security, privacy, pricing, and end-user aspects. We discuss the key enablers and challenges, and identify the key research questions for the development of intelligent edge services. As the main outcome of this white paper, we envision a transition from the Internet of Things to the *Intelligent Internet of Intelligent Things and provide a roadmap for the development of the 6G intelligent edge*.

1 Introduction

Edge intelligence (EI), powered by artificial intelligence (AI) techniques (e.g. machine learning, deep neural networks, etc.), is already considered one of the key missing elements in 5G networks and will most likely resent a key enabling factor for future 6G networks in supporting their performance, new functions, and new services. Consequently, this whitepaper aims to provide an overarching understanding of why edge intelligence is an important element of 6G, and what the leading design principles and technological advancements are that are guiding the work toward 6G edge intelligence.

In recent years, we have witnessed a growing market and exploitation of AI solutions in a wide spectrum of ICT applications. AI services are becoming increasingly popular in various ways, including intelligent personal assistants, video/audio surveillance, smart city operations, and autonomous vehicles. Indeed, entire industries are taking new forms—a prime example being Industry 4.0, which aims to digitize manufacturing, robotics, automation, and related industrial fields as part of digital transformation. Furthermore, the increasing use of computers and software calls for new types of design tradeoff, concerning, for example, energy and timing constraints of computations and data transmissions, as well as privacy and security.

The increased interest in AI can be attributed to recent phenomena, high-performance yet affordable computing, and the increasing amount of data generated by various ubiquitous devices, from personal smartphones to industrial robots. Powerful and inexpensive processing and storage cloud-computing resources are available for anyone with a credit card, where the abundance of resources meets the hungry requirements of AI, calling for the elaboration of enormous quantities of big data. Furthermore, the **high density of base stations in megacities (and high density of devices) provides a good basis for edge and fog computing.**

The devices generating and consuming data are commonly located at the edge of networks, near the users and systems under monitoring, surveillance, or control. However, this megatrend has received only little attention. Indeed, the wide diffusion of smart terminals, devices, and mobile computing, the Internet of Things (IoT), and the proliferation of sensors and video cameras are generating several hundred observations and ZBs of data at the network edge. Furthermore, increasing use of machine-learning models with a small memory footprint—such as TinyML—that operate at the edge plays an important role. Taking this into account in computational models means the centralized cloud computing model needs to be extended toward the edge.

Edge computing (EC) is a distinguished form of cloud computing that moves part of the service-specific processing and data storage from the central cloud to edge network nodes that are physically and logically close to the data providers and end users. Among the expected benefits of edge-computing deployment in

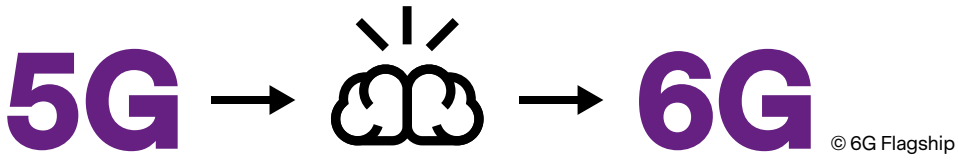


Figure 1: The transition from 5g to 6g enabled by edge intelligence.

current 5G networks are performance improvements, traffic optimization, and new ultra-low latency services. Edge intelligence in 6G will significantly contribute to all these aspects. Moreover, edge intelligence capability will enable the development of an entirely new category of products and services. New business and innovation avenues around edge computing and edge intelligence are likely to emerge rapidly in several industry domains. Sometimes, the term fog computing is also used to highlight that in addition to running things at the edge, computers located between the edge device and the central cloud are used. While various definitions, with subtle differences, for edge and fog computing exist, we use the terms interchangeably to denote flexible executions that are run in computers outside the central cloud.

A definition of particular importance for 5G and beyond 5G systems is given by the multi-access edge computing (MEC) initiative within ETSI[1]. In this architecture, a mobile edge host runs a mobile edge platform that facilitates the execution of applications and services at the edge. The ETSI MEC standard connects the MEC applications and services with the cellular domain through standardized APIs such as access to base station information and network-slicing support. From the data analytics perspective, edge intelligence refers to data analysis and the development of solutions at or near the site where the data is generated and further utilized. Edge intelligence thus allows the reduction of latency, costs, and security risks, making the associated business more efficient. From the network perspective, edge intelligence mainly refers to intelligent services and functions deployed at the edge of the network, probably including the user domain, the tenant domain, or close to the user or tenant domain, or across the boundary of network domains.

In its basic form, edge intelligence involves an increasing level of data processing and the capacity to filter information on the edge. However, intelligence is defined a priori. With increasing levels of artificial intelligence at the edge, it is possible to bring some AI features to each node, as well as clusters of nodes, so that they can learn progressively and possibly share what they learn with other similar (edge) nodes to collectively provide new value-added or optimized services. **Hence, it can be dicted that the evolution of telecom infrastructures toward 6G will consider highly distributed AI, moving the intelligence from the central cloud to edge-computing resources.** Target systems include advanced IoT applications and digital transformation projects. Furthermore, edge intelligence is a necessity for a world in which intelligent autonomous systems are commonplace, in particular when considering situations in which machines and humans cooperate (such as working environments) for safety reasons.

Software and hardware optimized for edge intelligence are currently in their infancy, and we are seeing an influx of edge devices such as Coral[2] and Jetson[3] that are capable of performing AI computation. Regardless, current AI solutions are resource- and energy-hungry, and time consuming. Indeed, many commonly used machine-learning and deep neural network algorithms still rely on Boolean algebra transistors to do an enormous amount of digital computations over massive-scale datasets. In future, the number and size of available datasets will only increase, whilst AI performance requirements will be increasingly stringent, for the expected (almost) real-time ultra-low latency applications. We believe this trend cannot really be sustainable in the long term.

To provide a concrete example of non-optimal hardware and software in 5G, we point out that in the basic functioning of especially deep neural networks (DNN), each high-level layer learns increasingly abstract higher-level features, providing a useful, and at times reduced, presentation of the features to a lower-level layer. An obstacle is that chipset technologies are not becoming faster at the same pace as AI solutions are progressing in serving markets' expectations and needs. Nanophotonic technologies could help in this direction: DNN operations are mostly matrix multiplication, and nanophotonic circuits can make such operations almost at the speed of light and very efficiently due to the nature of photons. Simply excessed, photonic/optical computing uses electromagnetic signals (e.g. via laser beams) to store, transfer, and process information. Optics has been around for decades, but it has until now been mostly limited to laser transmission over optical fiber. Nanophotonic technologies using optical signals to perform computations and store data could accelerate AI computing by orders of magnitude in latency, throughput, and power efficiency.
In-memory computing is a promising approach to addressing the processor–memory data transfer bottleneck in computing systems. In-memory computing is motivated by the observation that the movement of data from bit cells in the memory to the processor and back (across the bit lines, memory interface, and system interconnect) is a major performance and energy bottleneck in computing systems. Efforts that have explored the closer integration of logic and memory are variously referred to in the literature as logic-in-memory, computing-in-memory, and processing-in-memory. These efforts may be classified in two categories—moving logic closer to memory, or near memory computing, and performing computations within memory structures, or

[1] ***https://www.etsi.org/technologies/multi-access-edge-computing***
[2] ***https://www.coral.ai/***
[3] ***https://developer.nvidia.com/buy-jetson***

in-memory computing [1]. In-memory computing appears to be a suitable solution for supporting the hardware acceleration of DNN. System-on-chip architectures like the adaptive computing acceleration platform (ACAP) are yet another approach for AI applications. ACAPs integrate generic CPUs with AI- and DSP-specific engines, as well as programmable logic, in a single device. Internal memory and high-speed interconnection networks make it possible to implement the whole AI processing pipeline within a single device, eliminating the need to transfer data off chip [2].

Software supporting AI development is also an under-studied aspect of current 5G development. The tools, methods, and practices we use to build edge devices, cloud software, the gateways that connect them, and end-user applications are diverging for various reasons, including performance, memory constraints, and productivity. This means that the responsibilities of different devices are still largely defined a priori during their design and implementation. We are therefore far from software capabilities that would allow software to "flow" from one device to another ("liquid" software). **Without liquid software as part of future 6G networks, we are stuck with an approach in which we must decide where to locate the intelligence in the network topology at the due design time, because the computations cannot be easily relocated without design-time parations.**

In this white paper, we aim to shed light on the challenges of edge AI, potential solutions for these challenges, and a roadmap toward intelligent edge AI. In addition, another 6G white paper is written to highlight machine learning capabilities wireless communication networks [3]. This paper is structured as follows: In Section 2, we discuss the related work that motivates the paper; in Section 3, we provide an insight into our vision of edge AI; in Section 4, we address challenges and key enablers of edge AI in the context of the emerging 6G era; in Sections 5 and 6, we sent key research questions and a roadmap for meeting the vision.


Basel
Bern
Zürich
Luzern

2 Related work

Vision-oriented and positioning papers on 6G edge intelligence are starting to emerge. Zhou et al. [4] and Xu et al. [5] conduct a comprehensive survey of the recent research efforts on edge intelligence. Specifically, they review the background and motivation for artificial intelligence running at the network edge, concentrating on deep neural networks (DNN), a popular architecture for supervised learning. Further, they provide an overview of the overarching architectures, frameworks, and emerging key technologies for the deep learning model for training and inference at the network edge. Finally, they discuss the open challenges and future research directions in edge intelligence.

Rausch and Dustdar [6] investigate the trends and possible "convergence" between humans, things, and AI. In their article, they distinguish three categories of edge intelligence use cases: public, such as smart public spaces; private, such as personal health assistants and predictive maintenance (corporate); and intersecting, such as autonomous vehicles. It is unclear who will own the future fabric for edge intelligence, whether utility-based offerings for edge computing will take over as is the case in cloud computing, whether telecommunications will keep up with the development of mobile edge computing, what role governments and the public will play, and how the answers to these questions will impact engineering practices and system architectures.

To address the challenges of edge intelligence data analysis, computing power limitation, data sharing and collaborating, and the mismatch between the edge platform and AI algorithms, Zhang et al. [7] introduce an open framework for edge intelligence (OpenEI), which is a lightweight software platform to equip the edge with intelligent processing and data-sharing capability. Similarly, the ARM compute library[4], the Qualcomm Neural Processing SDK[5], the Xilinx Vitis AI[6], and Tensorflow lite[7] offer solutions for performing AI computations on low-power devices that can be deployed at the edge. More generally, the experience of edge computing is worth recalling. Mohan [8] adopts an edge-computing service model based on a hardware layer, an infrastructure layer, and a platform layer to introduce a number of research questions. Hamm et al. [9] present an interesting summary based on the consideration of 75 edge-computing initiatives. Edge Computing Consortium Europe (ECCE)[8] aims to drive the adoption of the edge-computing paradigm within manufacturing and other industrial markets with the specification of a reference architecture model for edge computing (ECCE RAMEC), the development of reference technology stacks (ECCE edge nodes), the identification of gaps, and the recommendation of best practices by evaluating approaches within multiple scenarios (ECCE pathfinders).

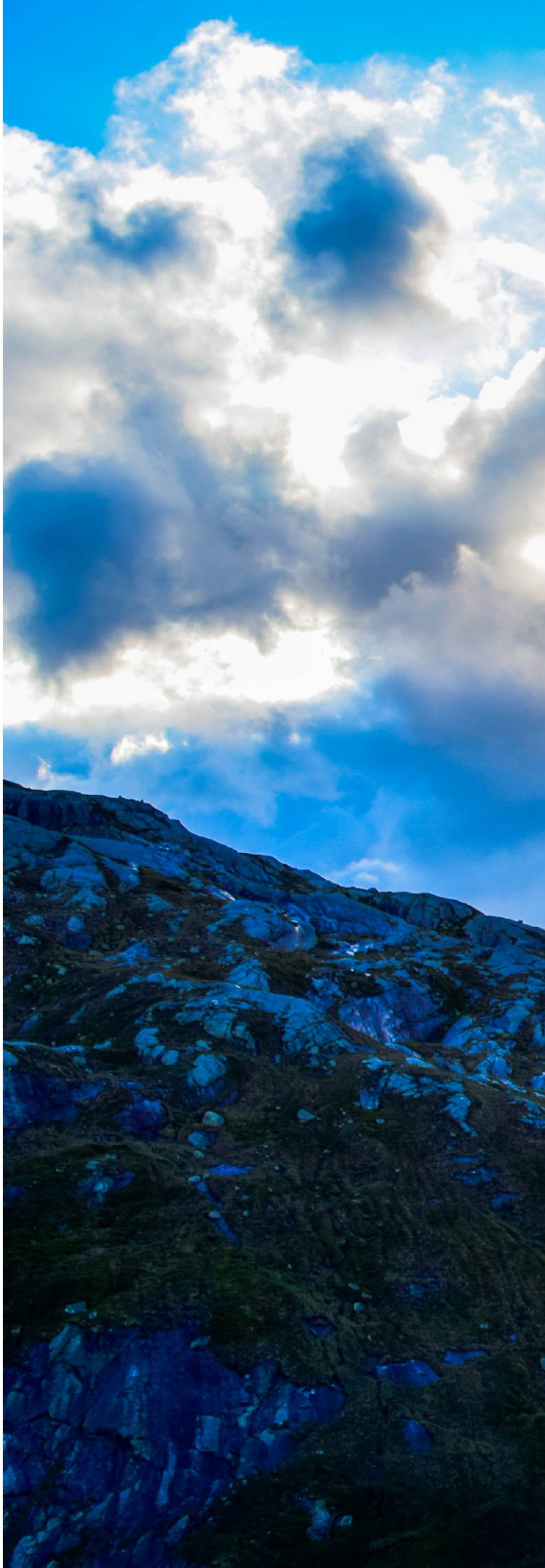
On the theoretical side, Park et al. [10] highlight the need for distributed, low latency, and reliable machine learning at the wireless network edge to facilitate the growth of mission-critical applications and intelligence devices. Therein, the key building blocks of machine learning at the edge are laid out by analyzing different neural network architectural splits and their inherent tradeoffs. Furthermore, Park et al. [10] provide a comprehensive

[4] *https://developer.arm.com/ip-products/processors/machine-learning/compute-library*
[5] *https://developer.qualcomm.com/software/qualcomm-neural-processing-sdk*
[6] *https://www.xilinx.com/products/design-tools/vitis/vitis-ai.html*
[7] *https://www.tensorflow.org/lite/*
[8] *https://ecconsortium.eu/*

analysis of theoretical and technical enablers for edge intelligence from different mathematical disciplines, presenting several case studies to demonstrate the effectiveness of edge intelligence for 5G and beyond.

In a series of position papers, Lovén et al. [11] divide edge AI into edge for AI, comprising the effect of the edge-computing platform on AI methods, and AI for edge, comprising how AI methods can help in the orchestration of an edge platform. They identify communication, control, security, privacy, and application verticals as the key focus areas in studying the intersection of AI and edge computing and outline the architecture of a secure privacy-aware platform that supports distributed learning, inference, and decision making by edge-native AI agents.

Almost identically to Lovén et al. [11], Deng et al. [12] separate AI for edge and AI on edge. In their study, Deng et al. discuss the core concepts and a research roadmap to build the necessary foundations for future research programs in edge intelligence. AI for edge is a research direction focusing on providing a better solution to the constrained optimization problems in edge computing with the help of effective AI technologies. Here, AI is used to enhance edge with more intelligence and optimality, resulting in Intelligence-enabled edge computing (IEC). AI on edge, on the other hand, studies how to carry out the entire lifecycle of AI models on edge. It is a paradigm of running AI model training and inference with device–edge–cloud synergy, with the aim of extracting insights from massive and distributed edge data with the satisfaction of algorithm performance, cost, privacy, reliability, efficiency, etc. It can therefore be interpreted as artificial intelligence on edge (AIE).



Vision for the 2030s edge-driven artificial intelligence

3

There is virtually no major industry in which modern artificial intelligence is not already playing a role. This is especially true in the past few years, as data collection and analysis have ramped up considerably thanks to robust IoT connectivity, the proliferation of connected devices, and ever-speedier computer processing. Regardless of the impact artificial intelligence is having on our present lives, it is hard to ignore that in the future it will enable new and advanced services for: (i) transportation and mobility in three dimensions; (ii) manufacturing and industrial maintenance; (iii) healthcare and wellness; (iv) education and training; (v) media and entertainment; (vi) ecommerce and shopping, (vii) environmental protection; (viii) customer services. The complexity of the resulting functionalities requires an increasing level of distributed intelligence at all levels to guarantee efficient, safe, secure, robust, and resilient services.

As with the transition we are experiencing from cloud to cloud intelligence, we are constantly assisting in an evolution from the "Internet of Things" to the "Internet of Intelligent Things." Given the requirements above, it is increasingly evident that an **"Intelligent Internet of Intelligent Things"** is also needed to make such an Internet more reliable, more efficient, more resilient, and more secure. **This is exactly the area where 6G communication with edge-driven artificial intelligence can play a fundamental role.**

Compared with edge-computing efforts from cloud service providers such as Google, Amazon, and Microsoft, there is a tighter integration advantage in computing and communication in 6G by telecom operators. For example, 6G base stations can be a natural deployment of edge intelligence that requires both computing and communication resources. This is likely to represent a new opportunity for telecom operators and to some extent, tower operators, to regain centrality in the market and increase the added value of their offer.

As the connected objects become more intelligent in the 6G era, it is difficult to believe that we can deal with them, the complexity of their use, and their working conditions by continuing to use the communications network in a static, simplistic, and dumb manner. The same need will likely emerge for any other services using future communications networks, including phone calls, video calls, video conferences, video on demand, and augmented and mixed reality video streaming, where the wireless communications network will no longer simply provide a "connection" between two or more people or a "video channel" on demand from a remote repository to the user's TV set, but will introduce a need to properly authenticate all the involved parties, guarantee the security of data fluxes eventually using a dedicated blockchain, and recognizing unusual or abnormal behavior in real time. Data exchange will in practice be

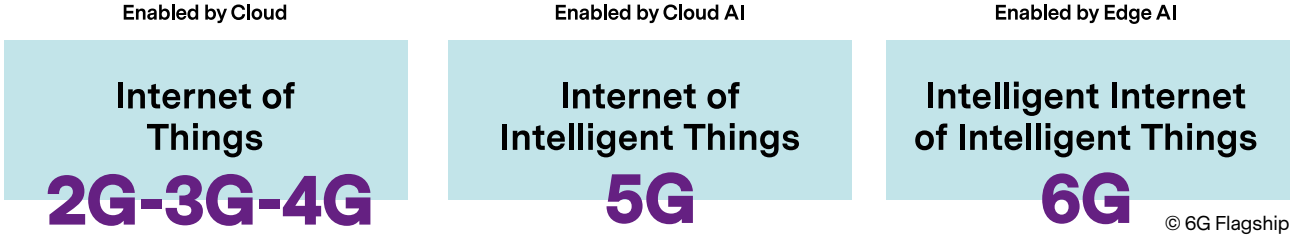


Figure 2: Evolution of the "intelligent internet of intelligent things"

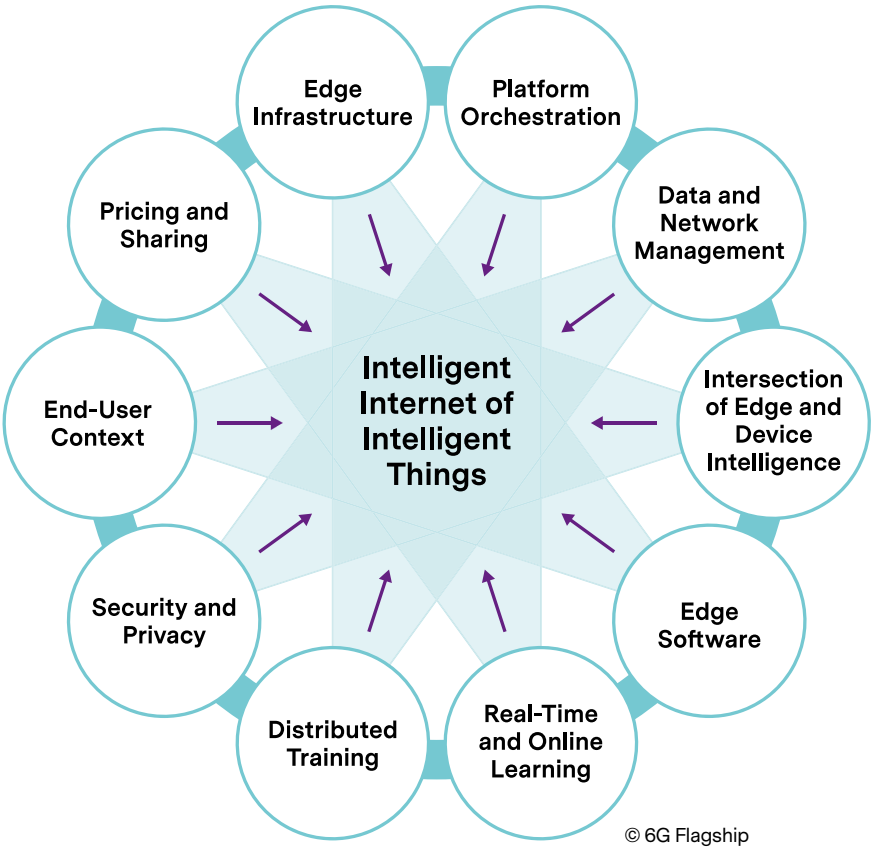


Figure 3: Key enablers for intelligent internet of intelligent things

much more than merely pure data exchange but will exchange a number of past, present, and possibly future properties of the data. In future 6G wireless communications networks, trust, service level, condition monitoring, fault detection, reliability, and resilience will define fundamental requirements, and artificial intelligence solutions are extremely promising candidates to play a fundamental role in satisfying such requirements.

We can easily anticipate that larger amounts of data will transit on the future 6G wireless communications network nodes, and a growing number of value-added applications and services will critically depend on this data. Bringing intelligence to the edge will clearly represent a basic functionality for guaranteeing the efficiency of future wireless communications networks in 6G, while representing the enabling technology for a number of value-added applications and services. **Artificial intelligence on wireless communications nodes can actually enable a number of advanced services and quality of service functionalities for the proposed applications.**

Existing computing techniques used in the cloud are not fully applicable to edge computing directly due to the diversity of computing sources and the distribution of data sources. Considering that even those solutions available to transform heterogeneous clouds into a homogeneous platform are not presently performing very well, Mohan [8] investigates the challenges for integrating edge computing ((i) constrained hardware, (ii) constrained environment, (iii) availability and reliability, (iv) energy limitations), proposing several solutions necessary for the adoption of edge computing in the current cloud-dominant environment. **Indeed, we define performance, cost, security, efficiency, and reliability as key features and measurable indicators of any AI for edge and AI on edge solutions.**

Zhou et al. [4] categorize edge intelligence in six levels, based on the amount and path length of data offloading. We extend Zhou et al.'s vision on edge-based DNNs to generic AI models and architectures, with seven levels where the edge can either be viewed as a set of single autonomous intelligent nodes, or as a cluster or collection of federated/integrated edge nodes. We also add a different degree of autonomy in the operation of the edge nodes (see Fig. 4). Specifically, our definition of the levels of edge intelligence is as follows:

- **Cloud intelligence:** training and inferencing the AI model fully in the cloud.
- **Level 1: Cloud-edge co-inference and cloud training:** training the AI model in the cloud but inferencing the AI model in an edge-cloud cooperation manner. Here, edge-cloud cooperation means that data is partially offloaded to the cloud.

- **Level 2: In-edge co-Inference and cloud training:** training the AI model in the cloud but inferencing the AI model in an in-edge manner. Here, in-edge means that the model inference is carried out within the network edge, which can be realized by fully or partially offloading the data to the edge nodes or nearby devices in an independent or coordinated manner.
- **Level 3: On-device inference and cloud training:** training the AI model in the cloud but inferencing the AI model in a fully local on-device manner. Here, on-device means that no data is offloaded/uploaded.
- **Level 4: Cloud-edge co-training and inference:** training and inferencing the AI model both in the edge-cloud cooperation manner.
- **Level 5: All in-edge:** training and inferencing the AI model in the in-edge manner.
- **Level 6: Edge-device co-training and inference:** training and inferencing the AI model in the edge-device cooperation manner.
- **Level 7: All on-device:** training and inferencing the AI model in the on-device manner.

Both AI for edge and AI on edge can be distributed at edge level. In practice, an edge node appears as a "local cloud" for the connected devices, and a "cluster of edge nodes" can cooperate to share the knowledge of the specific context and the specific environment, as well as to share computational and communication load, both during training and during inferencing.

Further, we list and summarize a number of key functions that we envisage as useful for possible future edge intelligence applications at all possible levels of Fig 4. Therein, we highlight where exactly the intelligence is "concentrated," and the applications and the services are "executed," depending on the specific application scenarios, the local environment, the network architecture, the cooperative framework that can be defined, and the performance and the costs that need to be balanced. Some examples of artificial intelligence methods to optimize telecom infrastructure in the 6G era and manage the lifecycle of edge networks (AI for edge) are recalled in Table 1; the edge as a platform for applica-

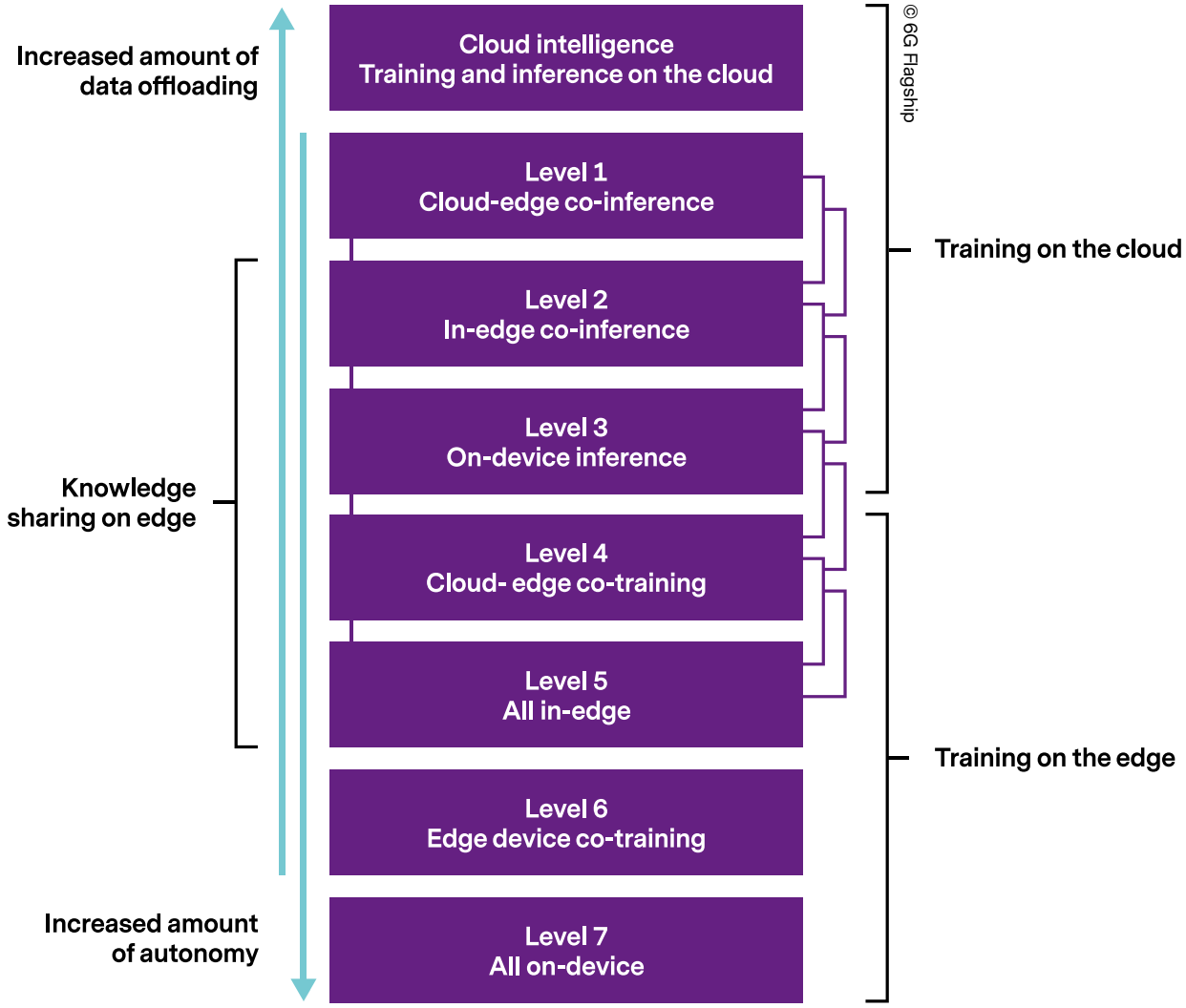


Figure 4: Level rating for edge intelligence (adapted from zhou et al [4])

tion-oriented distributed AI services (AI on edge) are listed in Table 2.

It is noteworthy that to guarantee efficient, safe, secure, robust, and resilient 6G-based services, it is also important to reduce dependencies between AI for the edge and AI on edge services. Infrastructure and platform orchestration functionalities should guarantee their coexistence and their optimization if they coexist, but they do not necessarily require these services to be fully implemented at all times. To allow maximum flexibility, we should probably develop an "ontology for 6G connectivity" to shape all the possible combinations of "micro services" on the edge nodes.

Table 1: AI for edge service	Specific objective
Wireless networking	Zhu et al. (2018) [13] describe a new set of design principles for wireless communication on the edge with embedded machine-learning technologies and models, collectively named learning-driven communication. It can be achieved across the whole process of data acquisition, which are in turn multiple access, radio resource management, and signal encoding.
mmWave xhaul systems	Development of mmWave xhaul systems, including AI/ML-based optimization, fault/anomaly detection, and resource management. Small cells, cloud-radio access networks (C-RAN), software-defined networks (SDN), and network function virtualization (NVF) are key enablers for addressing the demand for broadband connectivity with inexpensive and flexible implementations. Small cells, in conjunction with C-RAN, SDN, and NVF, impose very stringent requirements on the transport network. Here, flexible wireless solutions are required for dynamic backhaul and fronthaul architectures alongside very high capacity optical interconnects, and AI to maximize the collaboration between the cloud and the edge can represent a key solution.
Communication service implementation	Edge intelligence can automate and simplify the development, optimization, and run-time determination of communications service implementation. Edge intelligence in this case enables/assists service execution by determining the optimal/possible execution of service, based on the resource availability in a network.
Dynamic task allocation	Offloading and onloading computational tasks and data between participating devices, edge nodes, and cloud, in addition to smart and dynamic (re-) allocation of tasks, could become the hottest topic in AI for edge. Dynamic task allocation studies the transfer of resource-intensive computational tasks from resource-limited mobile devices between the edge and the cloud, and the interoperability of local devices sharing their computational power. These processes involve the allocation of various different resources, including CPU cycles, sensing capabilities, available data and AI models, and channel bandwidth. AI technologies with strong optimization and communication abilities can therefore be used extensively in the 6G era.
Liquid computing handover	A seamless handover can be further extended to cover the handover of the tasks being shared between devices and edge nodes while devices move in the network.
Location-based optimization	The optimization of network coverage and wireless networking can greatly benefit from the information collected progressively on the local radio environment. Basically, in this case, the devices exploit the knowledge available on the environment and the edge nodes.

Table 1: AI for edge service	Specific objective
Location-based optimization	Quality of Service can be extended by predicting the behavior of the devices interacting with a specific edge node or group of edge nodes. Information on the behavior and on the QoS can be shared between edge nodes (with due attention to possible privacy and security issues). In this case, the edge nodes essentially exploit the knowledge available on the environment and the devices in that environment.
Energy management	Although energy management for mobile devices has experienced significant improvements from the hardware perspective, we observe a high variability of devices' energy performance in connection with the applications. With the increasing level of autonomy that we expect for devices and 6G communications networks, we need to further extend the energy efficiency and energy management capacity for both 6G devices and 6G edge nodes.

Table 2: AI on edge service	Specific objective
Novel application areas	Autonomous and driving-assisted vehicles, autonomous drones, traffic control, smart factories, smart farms, smart roads, smart homes, and smart cities, can actually define the reference profiles for services and for wireless communications network functionalities to be activated.
Data intelligence	Edge Intelligence is to use advanced communications technology and AI to support ubiquitous data collection, aggregation, fusion, processing, distribution, and services at the edge. The ability to learn, infer and control from data, in both static and dynamic environments, is an additional value-added feature.
Cooperative intelligence	Algorithms run on heterogeneous platforms which may be geographically distant (imposing latency requirements) jointly solving an AI problem.
Real-time requirements	Localized AI/ML functions with constrained computation resources and (usually) strict real-time requirements.
Computing as a service	Provide intelligent computing capabilities when and wherever the user needs them (satisfying his/her requirements in terms of computing power, latency, energy consumption, cost, mobility, service reliability, etc.).
Advanced IoT models	IoT data models, architectures, and smart services, especially with distributed services in IoT over different platforms and implementations. Here, the focus is on enabling services that can adapt, based on the available IoT devices/services in the network. Connected "smart objects" operate in an intelligent virtualized computational environment that is deployed across cloud, edge, and mixed layers, vertically and horizontally.

4 Challenges and key enablers

Although the benefits of edge intelligence are immense, the realization of the intelligence (training) and the focus of applications (inference) pose several technical challenges, in contrast to traditional centralized artificial intelligence systems. It is therefore crucial to identify and analyze these challenges in edge intelligence and seek novel theoretical and technical enablers. In this regard, a set of prominent challenges in edge intelligence and some key enablers for overcoming them are discussed next.

Edge infrastructure solutions

Edge-computing infrastructures are best exemplified by the MEC reference architecture, currently under standardization by ETSI. The architecture describes edge platform components, their roles and expected functionalities, system APIs, and interactions for collaboration and third-party software integration. The target is an open multi-vendor edge platform. Guidelines on how to realize systems, applications based on the reference architecture, and a set of proof-of-concept applications are therefore presented. However, the implementation details of the system components and interactions are left open, and the architecture is based on distributed operation and control at two levels: system-level management and host-level management. However, the centralized orchestrator component is expected to have sole authority for all system resources. Platform- and

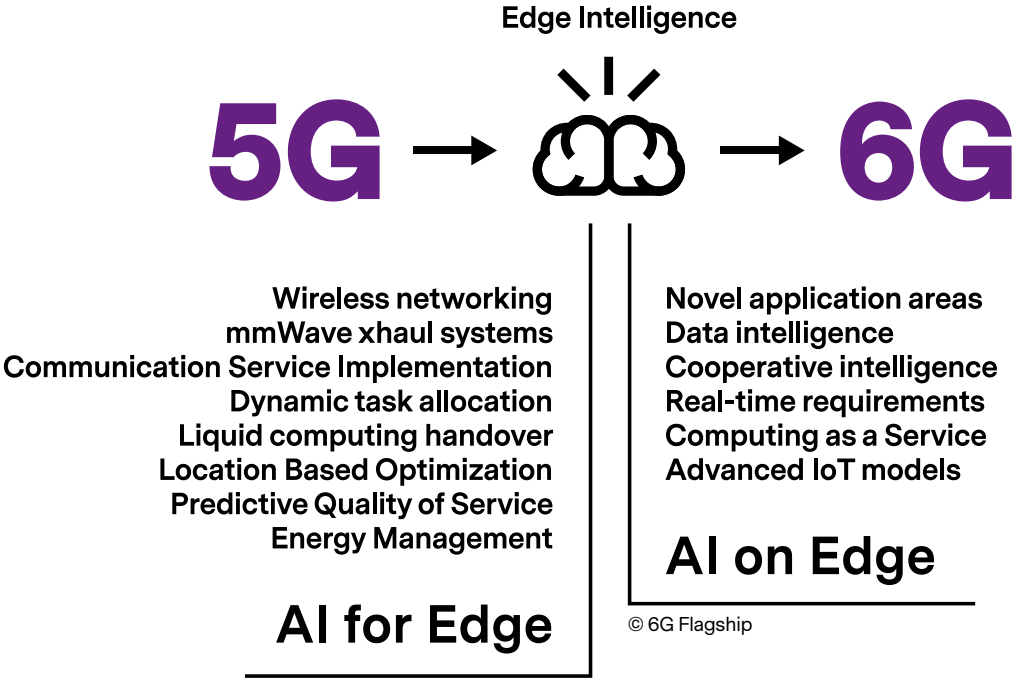


Figure 5: Key challenges and key enablers for edge intelligence for 6G

host-level components operate based on instructions received from the orchestrator with partial autonomy to control the resources under their domain. These components are expected to provide feedback to orchestrators about their operation. These operational principles are certainly beneficial, but they lead to challenges in real-time reactivity, providing low latency for multi-tenant applications, data routing, and aggregation system information delivery, etc. across dynamic and opportunistic distributed IoT environments. **Regarding AI capabilities for edge infrastructures, processing the system-wide data of resource usage and sharing across the deployment, system performance in relation to key performance indicators (KPIs), application data delivery, QoS, and Quality of Experience (QoE) parameters, etc. used for building models, learning, and further making predictions for the optimization of system behavior is largely unexplored territory.**

From the architectural perspective, different approaches can be found. ETSI MEC is a two-tier architecture [14], with management components in the cloud or platforms with similar capabilities, and application components deployed in the network edge layer below. In turn, fog computing provides a hierarchical computing platform across the deployment [15], typically confined in a space where the computational units, i.e. fog nodes, have increasing capacity toward the cloud. Cloudlets are a similar concept, in which on-demand physical computational capacity can be deployed in location as server racks and under-/overused capacity shared across the platform, with the additional cost of moving application components [16]. In such a multi-tier environment, the role of autonomous management and the operation of local components are even more important and challenging. Recently, mist computing has emerged, in which the data producing IoT devices such as WSN nodes and mobile devices is already harnessed for application-specific data processing at the data source. Another approach for device-level computing is mobile (cloud) computing, where the UEs of users provide ad hoc shared computational capabilities at a location with offloading to the edge/cloud. In addition to MEC standardization, open-source solutions for edge platform management such as Google Kubernetes and Docker Swarm exist that are widely used in industry.

Edge platform orchestration

The opportunistic nature of the IoT environment and large physical scale of edge-computing systems justifies AI approaches for the orchestration and management of such systems. Further, optimization toward fulfilling the edge promises, e.g. efficient resource use and QoE, requires a large set of different data sources and complex data analysis algorithms. Such centralized algorithms would be initially difficult to design and develop, and later deploy, maintain, and evaluate. Moreover, distributed and partitioned edge application execution is well aligned with the underlying architecture. A common challenge is to address the resource allocation problem, i.e. where to physically deploy edge-computing infrastructure, and what capabilities are needed in each location and its supporting logical "neighborhood" atop the physical network topology. A well-known approach here is to deploy the component next to the existing infrastructure, e.g. wireless access points following the existing underlying network topology, or harness location-based low-resource computational units, as in fog computing. Such deployments are limited by budgets and edge device capabilities, and thus need to be carefully pre-planned, e.g. based on historical data, but also in response to online and predicted application workloads.

Data and network management

Availability, accessibility, and types of data play pivotal roles in edge intelligence. In contrast to the conventional centralized artificial intelligence, the concept of edge intelligence in most circumstances relies on "small data." Hence, generalizing the edge intelligence reliably over unseen data is a critical challenge. On the other hand, even when edge devices have a considerably large fraction of data, it is crucial to identify duplicates and anomalies to refine rich data in avoiding performance losses (e.g. due to overfitting) in artificial intelligence models. Furthermore, applications relying on edge intelligence may generate different types of data with multi-sensory (audio, video, haptic), spatial, temporal, and stochastic characteristics. Additionally, because this heterogeneous sensory data is aggregated over a large network, the data itself may have inconsistencies. Thus, the fusion of these heterogeneous data types affects edge intelligence performance.

In addition to data, network states and requirements may change over time, even with extremely short durations (mission-critical applications) demanding tight response times. Under such changes, trained artificial intelligence models need the capability of adapting or coping mechanisms. Sharing fractions of data and trained AI models instead of raw data will significantly reduce the communication payload size over any network, increasing the potential size of the data systems that can cope in a comparably short period. **It is therefore crucial to understand and define data and model provenance and lifecycle well and provide measures to compare models and their fit to a current context.**

Pre-processing data for machine learning is beneficial to govern efficient and reliable artificial intelligence models in edge, cloud, and remote centralized data centers. Edge devices with large volumes of data can use clustering techniques to identify similarities therein. They can also use the tools of anomaly detection methods to isolate data inconsistencies. After the above classifications,

down-sampling techniques can be adopted to manage the ensuring of generalized edge intelligence. For small data issues, edge devices need to increase sampling frequency with the cost of energy consumption to obtain rich datasets. Additionally, synthetic data can be generated by resorting to well-trained generalized adversarial networks. Enabling incremental learning methods to train high-quality models over time and adopting formally trained models via knowledge distillation and transfer-learning techniques are promising solutions for coping with the issues posed by small data availability. Edge intelligence that needs to cope with the fusion of heterogeneous data types can utilize feature extraction techniques (i.e. representation learning) and split learning over multiple modalities. **To address the inconsistencies in heterogeneous sensory data, edge intelligence can resort to generalized adversarial networks, in which synthetic data can recover and restore the data consistency.**

Edge AI can be used to optimize the operations and performance of edge networks. However, in certain cases like fault detection and recovery, the problem is that failures are usually quite rare, and data for machine learning is thus very imbalanced. In general, network traffic and events tend to have self-similar behavior, which means that e.g. traffic anomaly detection mechanisms should be able to cope with heavy-tailed distributions. If user devices and applications start to adopt greedy AI-based flow control and path/GW selection mechanisms, the network traffic flow may become even more difficult to predict. While several techniques can be used to generate artificial data, it is not necessarily evident that such data is optimal for training, e.g. in fault detection systems. An alternative is to use simulations to obtain sufficient amounts of balanced data, which requires a kind of edge network digital twin. Furthermore, as the used RF frequencies are moving towards higher mmW bands (and over), it may also be beneficial to integrate other types of data with network management, like fine-grained localized weather information (heavy rain events) and seasonal changes in foliage. **To summarize, it would be better to understand the behavior of the network instead of handling it like a black box.**

Intersection of the edge and device intelligence

The location of the edge application has the utmost importance for real-time reactivity and adaptivity in response to the dynamic environment and user movement. To address the extreme end of the distributed edge, edge-supported approaches including mobile devices as part of the computational platform have emerged. These include mobile cloud computing, mobile edge, and mobile fog. Naturally, such platforms at the low end possess limited capabilities for “small data” processing, analysis, and dissemination, where further support from the edge is required for advanced analysis. Therefore, such distributed applications are typically partitioned with software components on the user devices, and edge and cloud layers. Here, sharing of (refined) data and local resources both horizontally and vertically has the utmost importance for saving device resources, i.e. energy, in the participating devices and providing operational capacity in response to user mobility. The key challenge is interoperability, i.e. uniform interfaces, to share data, results, tasks, and high-level AI models (e.g. algorithms). **Moreover, the massive scale of such distributed deployments across networks significantly increases the scale of management and orchestration with a holistic view of system operation, introducing further latencies to the control.** Lightweight AI solutions are therefore already needed at the mobile device level to increase autonomy and self-capabilities*.

Here, a classical distributed AI paradigm, software agents, and multi-agent systems have shown benefits in providing autonomy, reactivity, adaptivity, machine learning, code mobility, and collaboration capabilities [17, 18]—even for resource-constrained IoT devices [19]. Such devices are commonly known as smart objects in the IoT context [20, 21]. Existing use cases for agents in the cloud–edge–device continuum include representing system entities, facilitating collaboration both horizontally and vertically, sharing of resources, and controlling (e.g. SDN) and monitoring system operation, networks, and devices. **However, increasing the agent capabilities from reactive operation toward deliberative agents with cognitive capabilities, e.g. learning and proactivity, is an open question.** Further AI techniques, facilitating both vertical and horizontal collaboration and cooperation, include swarm intelligence, game theory, and genetic algorithms.

Software development for edge

Software development for edge systems relies on virtualization, exemplified by virtual machines and lightweight containers. Edge applications are developed as software packages, possibly implemented by multiple stakeholders, from where the application images are automatically built by edge system management components that maintain the application lifecycle. The images are then deployed to the edge hosts atop the virtualization infrastructure, according to the system policies. They are further managed and instantiated by platform and host components, which are also responsible for providing the required application-specific service, data, and network access and maintaining the required QoE. In addition to deployment considerations, the challenge here is to manage system policies, SLAs, access rights, billing, etc., for the software packages, and negotiate and orchestrate their use online, possibly with external third-party service providers and network operators. In deploying and launching edge applications, both push

and pull approaches are facilitated by offloading from UE and pulling application-specific components close to the infrastructure. **Toward autonomy, the horizontal code migration of selected software component in a limited scope is also an enabler.** In this context, security and resources, as well as information sharing, are important challenges.

Virtual machines typically provide monolithic self-contained application images, typically of several gigabytes, that become resource consuming to deploy and move across the edge platform. Microservices are a distributed approach for edge application development. Their goal is to develop in isolation modular application components at the individual process level, which can be individually deployed on demand to build the application workflow. **Here, a lightweight version of virtual machines and containers encapsulates individual microservices for deployment.** Unikernels that can run as virtual machines or even at bare metal provide an alternative to containers. The image size of unikernel-based microservices can be more or less the same size as container-based alternatives, so unikernels could provide better isolation with the same resources.

DevOps practices such as CI/CD provide ways of developing microservices in isolation, managing versioning, and deploying such components automatically using system management components. Obviously, managing such large-scale automatization online is challenging, because the additional small-scale application-specific units and their workflows significantly increase the scope of system and package management, and related performance monitoring. There are some AI-specific CI/CD frameworks like Kubeflow[9] and MLFlow[10] that support AI model development, training, and deployment workflows in cloud environments. However, in edge AI environments, such frameworks would need adaptation to edge data sources and federated cloud environments.

Real-time requirements and online learning

Novel and future AI applications require real-time feedback to be effective and address the challenges set by many real-world applications, including robotics and self-driving cars, traffic and logistics management systems, and telepresence, virtual, and augmented reality applications, all of which are included in 6G verticals and application areas. Real-time challenges cannot be solved only by decreasing latency and increasing network bandwidth because of the time usually spent on collecting the data for machine-learning models, training these models, and defining actions based on the learned models to be returned to the application. **Thus, redefining the entire real-time feedback cycle becomes even more crucial, including balance between pre-trained and online learned models, efficient model distribution and reutilization during their lifecycle, and dynamic decision making based on all the knowledge available from different models and data sources.**

To address the challenges due to the need for a short response time, it is essential to quickly adapt data dissemination and model training along the network changes, as well as to reduce the processing complexity in the inference. By using the frameworks of transfer learning and knowledge distillation, edge intelligence can reduce retraining latency with the aid of pre-acquired intelligence. Furthermore, knowledge distillation and model pruning allow the reduction of artificial intelligence models, yielding fast inference. In addition to the aforementioned methods, the dynamics in the data and the network can be addressed by resorting to reinforcement learning and the co-design of communication, control, and machine learning [22].

Developing distributedly trained algorithms

Toward realizing edge intelligence, the training procedure directly affects the majority of the end-to-end latencies, the inference reliability, and the overall scalability [11]. While a handful of applications may allow traditional centralized artificial intelligence model training and download a trained model for the inference at the edge, **the majority of mission-critical and privacy-concerned applications demand online distributed training algorithms that can be employed at edge devices.** From this perspective, on-device limitations and the communication bottleneck among edge devices, and between the edge and the servers, play a critical role in developing the distributed algorithms. The edge devices in a large-scale artificial intelligent system are likely to be mobile and thus powered with capacity-limited batteries and storage. The limited energy budget is used for both computation (training and inference) and communication.

While large machine-learning models and frequent coordination among edge devices are ferred for higher inference accuracy and reliability [23], they could be inefficient from the energy consumption perspective, bounded by storage/memory limitations and privacy requirements. These on-device constraints call for energy-efficient, low-complexity and low-capacity, privacy-sensitive designs of distributed algorithms. Under

[9] *https://www.kubeflow.org/docs/started/kubeflow-overview/*
[10] *https://mlflow.org/*

limited data availability, edge devices may require exchanging raw data itself, its model parameters, or inferred outputs/decisions among one another or with a central server to improve the reliability and robustness of the distributed algorithms. This coordination within the network suffers with the uncertainties in the communication links and the network dynamics. **Therefore, user and resource (computation and communication) scheduling, as well as data and model comssing, need to be accounted for in the distributed algorithm design.**

The aforementioned limitations of device capabilities and communication in distributed algorithm design can be addressed with several technical and theoretical enablers, listed below. Within the limited power and memory of devices, it is suitable to seek data and model parallelization techniques, depending on the privacy requirements. Here, data can be split into several batches and processed utilizing mobile edge-computing servers. Alternatively, large artificial intelligence models can be split over several devices in which sequential/parallel training can be carried out, i.e. adapting split learning within distributed training algorithms [24]. Additionally, depending on training data size and privacy requirements, as well as the artificial intelligence model size, federated learning, knowledge distillation, and transfer learning methods can be selected as model training techniques [25, 26, 27].

To improve communications efficiency within distributed training algorithms, the uplink-downlink channel capacity asymmetry in the wireless network can be exploited by jointly adapting knowledge distillation and federated learning. Moreover, artificial intelligence model pruning, and coded and quantized model/data transmission-based learning can be adopted to address the limitations of both communication and storage [28]. **Furthermore, it is important to identify the characteristics of the network dynamics when designing distributed training algorithms.** Since static algorithms may yield poor performance under drastic network changes, it is crucial to either introduce cold-start mechanisms to retrain the models or to adopt continuous knowledge acquisition via continual learning methods, including transfer learning, online learning, and reinforcement learning. In addition to the aforementioned technical enablers, it is mandated to ensure latency, reliability, and scalability guarantees within the distributed training algorithms. From the perspective of reliability, generalization error is a performance measure of a trained artificial intelligence model over unseen data. For distributed algorithms, the frameworks of meta distribution, risk management, and extreme value theory can be used as theoretical enablers to analyze and minimize the generalization errors. To reduce latency while developing secure and scalable distributed algorithms, tools from differential privacy, rate-distortion theory, and mean-field control theory can be adopted.

Security and privacy

In any edge-cloud computing environment, data is transmitted from the edge to the dedicated computing infrastructure with services that perform the data analysis, which can be either private or public. Since the data leaves the edge, it can be exposed to various vulnerabilities and attacks such as penetration attacks resulting in theft of information or even denial of service attacks, resulting in crashing servers or networks. Additionally, not only can an attacker access and intercept the data, but the application's processing outcome in transit can lead to a different action/scenario than the intended one (i.e. tampering). **The locality on the edge, as well as the potential proximity of the system to end users, can also enable it to help address certain security challenges.** In some applications, edge AI can also be used to improve security and privacy, e.g. by anonymizing human faces in video streams or replacing people with straw figures if the main application depends only on the number of people in a certain location, or if someone has fallen and is lying on the ground. The 6G white paper on security and privacy discusses these topics further [29].

In this view, lightweight and distributed security mechanism designs are critical to ensure user authentication and access control, model and data integrity, and mutual platform verification for edge intelligence. It is also important to study novel secure routing schemes and trust network topologies for edge intelligence service delivery when considering the coexistence of trusted edge nodes with malicious ones. On the other hand, end users and devices would generate a massive volume of data at the network edge, and this data may be privacy-sensitive, because it may contain the user's location data, health or activity records, manufacturing information, etc. Subject to the privacy protection requirement, i.e. the **EU's General Data Protection Regulation (GDPR)**, directly sharing the original datasets among multiple edge nodes can carry a high risk of privacy leakage. Thus, federated learning may be a feasible paradigm for privacy-friendly distributed data training, such that the original datasets are kept in their generated devices/nodes, and the edge AI model parameters are shared. To further enhance data privacy, research efforts are increasingly devoted to utilizing the tools of differential privacy, homomorphic encryption, and secure multi-party computation in designing privacy-serving AI model parameter-sharing schemes.

End-user aspects

One of the main goals of edge computing, and a justification for edge intelligence, is to maintain the required QoE for users in terms of network connectivity and application execution improvements, and adaptation to the **dynamic environment and user mobility. A key challenge in optimizing edge systems for QoE is understanding**

the user's context (e.g. location-awareness), based on both large-scale analysis of user behavioral patterns across the deployment and real-time reactivity in the local environment (e.g. in relation to network connectivity and latencies, bandwidth availability, data transmission and application execution requirements, and the integration of user-specific third-party components). These concerns lead to online and on-demand adaptation of local edge resources that propagates across the deployment, both horizontally and vertically.

Challenges are introduced by user mobility, leading to user- and application-specific (virtualized) component movement and migration across edge deployments, and to management challenges in data and (stateful) application migration while minimizing handover latencies. Further, dynamic edge resources are shared between multiple users in multi-tenant fashion, raising privacy and security concerns. Regarding mist computing, sharing user equipment as part of data collection and the computational platform requires incentives to encourage participation, such as micropayments, defining data ownership(s) and policies for sharing, and GDPR-compliant privacy protection schemes. Approaches to resenting users in edge systems have already been proposed, e.g. digital twins and software agents with cognitive capabilities. Here, building trust between edge platforms and users is required for successful cooperation.

Pricing and sharing mechanism

In future 6G networks, AI-powered mobile edge devices will be enabled to share their communication, caching, computation, and learning resources (3C-L resources) to satisfy the quality of requirements (QoE and QoS) for 6G wireless applications, such as tactile Internet, virtual reality, and autonomous driving. Hence, an intelligent 3C-L resource-sharing framework remains in its infancy and should be significantly addressed. All the resources can be shared by mobile edge devices to maximize the resources' utilization by virtualizing all the resources into the virtual resource pool.

To cope with such a challenge, 3C-L resource sharing can be modeled by a dynamic pricing mechanism from an economic perspective, in which mobile edge devices are modeled as intelligent agents that can price, i.e. operate as brokers, or purchase 3C-L resources and consume services according to their own requirements. Accordingly, an economic sharing model should be established to make 3C resources and knowledge tradable by a market equilibrium approach. In particular, the multi-agent distributed learning approach may be developed to make the optimal price and resource allocation decisions, considering the different QoE and QoS requirements of 6G network applications, services, and systems. Moreover, how to smartly record the price and disseminate the revenue according to the proof of work among the distributed edge devices is also very important. **Smart contracts and distributed ledgers are expected to play an important role in fully unleashing the fairness, security, and activity of this ecosystem.** Therefore, designing the appropriate sharing and incentive mechanisms, as well as lightweight consensus protocols for edge intelligence, should elicit escalating attention.

Table 3 summarizes the defined key challenges and enablers.

Table 3:	Key challenges	Key enablers
Edge-computing infrastructure	· Supporting dynamically changing resources and configurations · Reliable feature deployment · Device mobility	· Container technologies · Virtualization · Isomorphic software architectures · Handover protocols and techniques
Edge platform orchestration	· Addressing the resource allocation problem. · Addressing the resource distribution problem. · Addressing dynamic allocation and distribution problems.	· Virtualization optimization. · Data intelligence algorithms. · Data analysis algorithms. · Multi-level and fully distributed dynamics optimization algorithms.

Table 3:	Key challenges	Key enablers
Data and network management	· Fusion of heterogeneous data types to optimize edge intelligence performance. · Understanding and definition of data and model provenance and lifecycle. · Comparing models and their fit to a given context. · Enabling incremental learning methods. · Addressing inconsistencies in heterogeneous sensory data. · Modeling and understanding network behavior and performance. · Common Interoperability practices' and standards' optimized high-level AI models. · Centralized system management and operation, limiting control latencies	· Extension of sensor and data fusion algorithm to support appropriate data and network management. · Data and network fault detection and identification. · Knowledge extraction and incremental learning algorithms · Knowledge-sharing solutions. · Lightweight AI solutions to increase autonomy and self-capabilities*. · Agents with cognitive capabilities, e.g. learning and proactivity. · AI techniques for collaboration and cooperation, adopting well-known paradigms such as swarm intelligence, game theory, and genetic algorithms.
Intersection of edge and device intelligence	· Common interoperability practices and standards and optimized high-level AI models. · Centralized system management and operation, limiting control latencies	· Lightweight AI solutions to increase autonomy and self-capabilities*. · Agents with cognitive capabilities, e.g. learning and proactivity. · AI techniques for collaboration and cooperation, adopting well-known paradigms such as swarm intelligence, game theory, and genetic algorithms.
Software development for edge	· Dynamic configurations · Security · Flexible deployment · Debugging capabilities in development time	· Container technologies · DevOps, CI/CD · Virtualization · Liquid software that can flow from one node to another
Real-time requirements and online learning	· Adapting along the network dynamics · The need of short response times	· Transferring learning, knowledge distillation, and reinforcement learning · Co-designing communication, control, and machine learning
Distributedly trained algorithms	· Reducing the cost of coordination among edge devices · Reducing generalized errors in trained models	· Federated learning, knowledge distillation, transfer learning · Model pruning coded and quantized machine learning · Meta distributions, extreme value theory, risk management framework

Table 3:	Key challenges	Key enablers
Security and privacy	· Guaranteeing the implementation of security and privacy strategies according to user needs and system requirements. · Guaranteeing the recognition of abnormal behavior according to user requirements and operator criteria.	· Allowing the application of the security strategy at all levels. Using physical-level solutions to increase security and trust. · Allowing the appropriate management of personal information ownership at all stages.
End-user aspects	· Understanding user context based on both large-scale behavior patterns	· Digital twins and software agents with cognitive capabilities

5 Core research questions

The emergence of IoT and the demand for responsiveness, privacy, and context-awareness are pushing intelligence to the edge and pulling the challenges and enablers listed in Table 3. For numerous domains that utilize and benefit from 6G, edge Intelligence can be outlined with reference to the services to enable "AI for edge" in Table 1 and the services to enable "AI on edge" in Table 2. Next, we provide an overview of the core research questions that will be tackled in the development of 6G edge intelligence.

Edge infrastructure

1. What architectural considerations are involved in application development of edge intelligence? Can flexible architecture and pipeline cater for different types of edge device? Can seamless vertical and horizontal collaboration in physical deployments atop existing network topologies be achieved?
2. How can edge-dedicated classification and taxonomy for the edge intelligence components be defined?
3. What are the impacts on network architecture regarding outer-network edge intelligence and in-network edge intelligence, including network service-, function- (NSF) level, and inter-NSF intelligence? How can suitable network protocols and interfaces for edge intelligence be developed?

System platform and stack

4. Which software development practices, quality assurance, and testing methods could be applied in edge intelligence-powered applications?
5. What are the DevOps aspects?
6. How can diagnostics and cooperative diagnostics for edge specific algorithms be enabled? How can multiple edge entities be involved in debugging in a decentralized setting?
7. How can formal verification tools and processes be incorporated wherever possible?

Data and network management

8. How can online learning with (possibly) non-stationary data, federated learning methods, and energy-efficiency be organized?
9. How can agent-based systems to enable real-world cognitive be designed?
10. How can computing be distributed among the different resources?

Intersection of edge and device intelligence

11. How can lightweight AI solutions be implemented to increase autonomy and self-capabilities?
12. How can agents with cognitive capabilities be included?
13. How can AI techniques, facilitating both vertical and horizontal collaboration and cooperation, and including swarm intelligence, game theory, and genetic algorithms be optimized?

Software development of the edge

14. How can flexible and interoperable software agents be developed?
15. How can reconfigurability and continuous deployment be guaranteed?
16. What solutions can be adopted for maximum virtualization capacity?
17. How can liquid software solutions that can flow from one node to another be guaranteed?

Real-time requirements and online learning

18. How can the capacity to transfer learning and provide knowledge distillation and reinforcement learning be guaranteed?

19. How can co-design communication, control, and machine learning be implemented?
20. How can model training along the network changes be quickly adapted?
21. How can the processing complexity in the inference be reduced?

Distributed training, algorithmic design, and deployment

22. What is the impact of accelerators, and how are they changing the way we approach distributed algorithms and ML design?
23. How can enough (and powerful) resources for ML at the edge (e.g. hardware acceleration, GPUs, etc.) be provided in an economically sustainable way?
24. How can the energy required for performing the computation under different scenarios be balanced? Can it be achieved by enabling computational power/storage capacity/power consumption on the edge?
25. How can distributed training, inference, and control be carried out in a communications-efficient, reliable, and scalable manner?
26. How can ML algorithms be factorized that can run partly on heterogeneous platforms? (Also, with heterogeneous computing infrastructural resources? For example, training deep neural networks (estimation of parameters)?

Security, privacy, and portability

27. How can security of data, ML model tempering, and protocols be assured? (New protocols must be invented that use TPMs in edge devices.)
28. How can operation be failing safe in the event of network or node failure?
29. How can data provenance at scale be ensured?
30. How can the privacy concerns of users and regulatory bodies be satisfied?

End-user concerns

31. How can QoE-related KPIs of edge standardization be addressed?
32. How can resources be shared among different end users?
33. How can user mobility be handled?
34. How can stateful application migration be handled?
35. What are the social-technological influences of edge intelligence?
36. What regulation or cultural issues is involved?
37. How then can the owner, developer, operator, tenant, and user of edge intelligence be motivated to define the relationships among them and the relationship with those in the network?

Pricing and sharing mechanism

38. How can the 3C-L resources be virtualized?
39. How can an economic sharing model by market equilibrium approach be established?
40. How can the dynamic pricing mechanism be designed?
41. How can 3C-L resources be smartly traded, and the revenue disseminated?
42. How can the appropriate sharing incentive mechanism be designed?

6 Prospective use cases

Edge Intelligence methods will provide novel business opportunities and technological solutions for various application fields, including but not limited to personal computing, urban computing, and manufacturing, facilitating their efficient, safe, secure, robust, and resilient wireless networking. Next, we highlight some example application areas for edge intelligence, and especially the Intelligent Internet of Intelligent Things.

Edge intelligence for autonomous driving

Autonomous driving is one of the broad research topics covering applications ranging from simple driver assistance systems (such as traffic sign recognition and lane-keeping assistance) to fully automated driving without human support. One of the specific and interesting use cases for autonomous driving is the concept of autonomous vehicle platooning, i.e. the coordinated movement of a group of driverless cars. This group of short-distance vehicles, typically trucks, forms a convoy led by a platoon leader, responsible for sending the steering information to the platoon members.

The data exchange between the platoon members and between the platoon and other devices can be realized through different wireless communications technologies (such as the dedicated short-range communications, DSRC, or cellular networks, Cellular-V2X, C-V2X). However, when the number of communicating cars increases, it may suffer from the prospective medium congestion and may not be able to fulfill the stringent requirement of 99.99% reliability. In this context, the approach to offloading some data to other bands is gaining interest. EI is foreseen here as the enabler for dynamic spectrum access, which should allow for fast and reliable processing of data generated within platoons and by all users on the road. A hierarchical structure of the database-oriented system supporting these operations in V2X communications is shown in Figure 6.

We consider a highway scenario in which multiple platoons of cars travel among other vehicles. We assume that platoon cars are autonomous, with their mobility controlled using the cooperative adaptive cruise control (CACC) algorithm.

The intra-platoon communications in a dynamically allocated band as a secondary system should not cause the degradation of any existing licensed service. To support vehicular dynamic spectrum access, database-oriented systems equipped with dedicated units for data processing and decision making will specify which frequency bands can be used for data transmission. We claim that various kinds of information will be stored in local (regional) and global databases. Edge intelligence functionality will be provided by the dedicated advanced processing units that are co-located with the base station or roadside units.

Edge intelligence for smart spaces

Smart spaces such as smart homes, smart campuses, smart offices, and smart hospitals are expected to contain a variety of networked devices and AI-driven in-network services to aid everyday activities. These devices and services will be latency-sensitive. They are expected to exchange privacy-sensitive information, and some of these devices, such as surveillance cameras, are expected to generate large volumes of data. Satisfying the requirements of applications that use these devices and services will require edge-native solutions. For example, data generated by the networked devices in smart hospitals may be privacy-sensitive. Regulations such as GDPR may mandate storing and processing of the data on the hospital mises. Similarly, AI services for object and face recognition from live surveillance footage in smart spaces will require real-time processing of large volumes of data, motivating the need for edge-native solutions.

Edge intelligence for environmental sensing

Environmental sensing such as air quality monitoring requires the collection and processing of large volumes of data from a variety of sensors sad across large geographical areas [27]. For example, accurate air quality monitoring demands a high spatial and temporal resolution of air quality data from sensors monitoring humidity, temperature, particulate matter concentrations, and gaseous pollutants. Collecting this data requires the dense deployment of sensors and real-time processing, and filtering of the raw analog data collected by the sensors. AI can be leveraged to identify the optimal locations for sensor deployment, the trajectories for the mobile sensors, the calibration of inexpensive sensors, and the locations for performing the computation. Edge-native solutions for processing

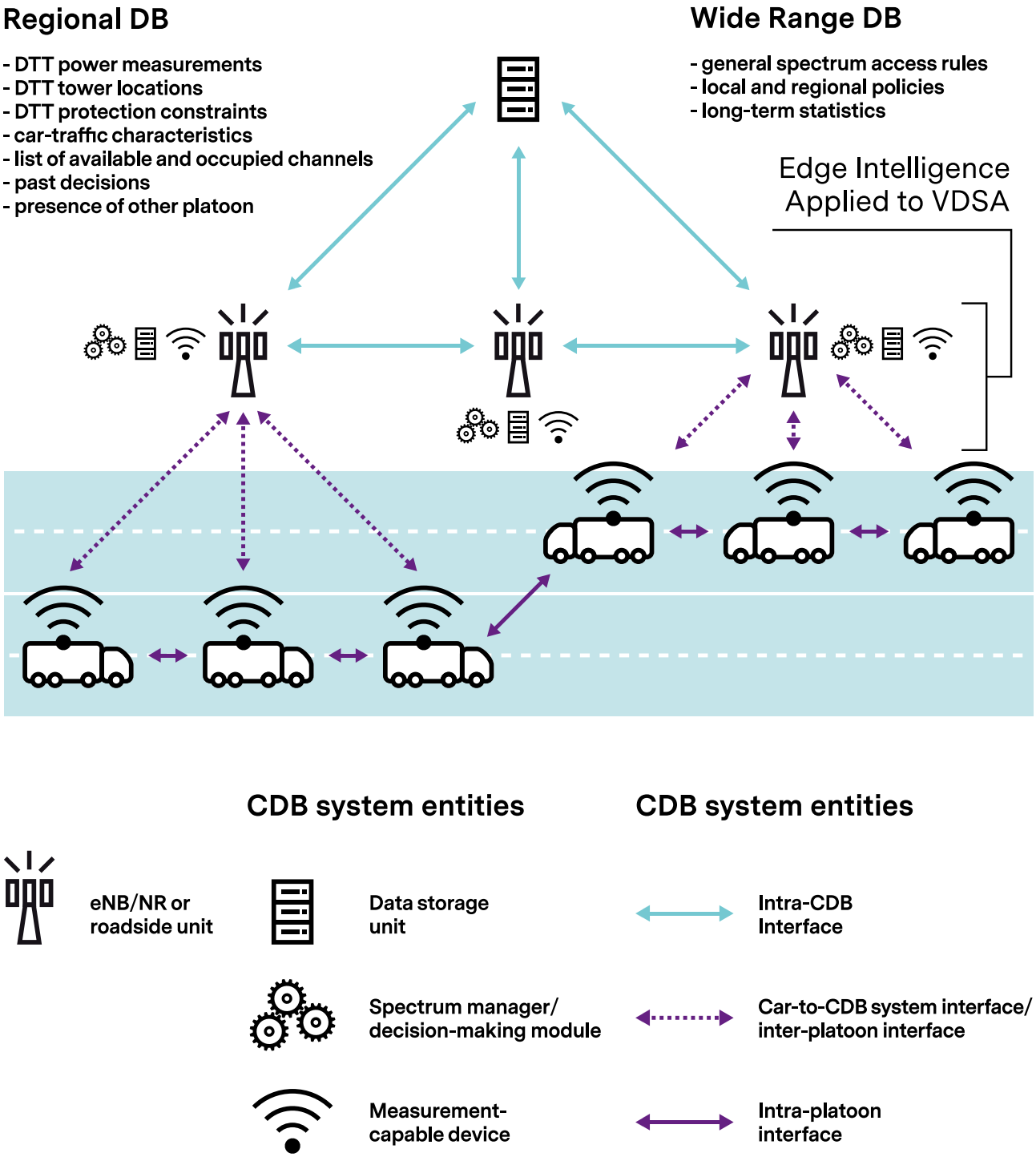


Figure 6: Edge intelligence applied to vehicular dynamic spectrum access in platoons

the raw data may help reduce the network load and parallelize the computation.

Edge intelligence for mobile XR

Mobile extended reality (XR), a portfolio encompassing all virtual or combined real-virtual environment compounds, including virtual reality (VR), augmented reality (AR), and mixed reality (MR), is a promising AI-powered 6G service application (e.g. the future tactile Internet). Visuo-haptic XR allows remote interaction with real and virtual elements (objects or systems) in perceived real time, driving massive real-time data at the network edge. It is therefore a kind of computation-intensive and data-craving application with low latency support. Edge intelligence has demonstrated great potential especially for XR service in resource and energy-constrained devices. To address the limitations of devices' battery energy and computation capacity, and reduce end-to-end latency, intelligent task segmentation, computing offloading, and learning model sharing will play a major role in bringing immersive experiences to users.

Next-generation cobots (collaborative robots) in manufacturing

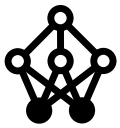
The collaboration between robots and humans in various domains will increase and become more seamless in the future. In manufacturing, not only are routine tasks being transferred from humans to robots (while upskilling people), but collaborative robots are also performing task-level collaboration with humans. In their current state, cobots are stationary devices with fixed gripper mechanisms and task programming. In future, cobots are envisioned with the following functionalities: automatic monitoring of machine health properties; autonomous or semi-autonomous navigation on the factory floor; switching from one workstation to another through task-level adaptation; and collaborating as a fleet. These functionalities will call for various sources of real-time data generation for cobots themselves, as well as low latency communications and tight collaboration with MES (manufacturing execution system) systems and factory private clouds. This again calls for a role for edge intelligence in performing fine-grained control of cobots, as well as coordinating larger production goals with the back-end MES/cloud.

Roadmap to the Edge Intelligence

5G

5G Edge

First commercial 5G MEC deployments

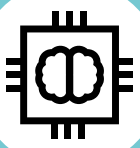


Pre-trained Edge

Pre-trained AI models used for processing data at the edge

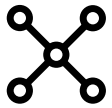
Edge AI

AI features brought to each edge node (ability to learn and to share models with other edge nodes).



Dedicated Hardware

Specialized edge devices capable of performing AI computation

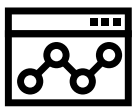


Distributed AI

AI algorithms distributed in a network of edge devices, providing low-latency and reliable results

Learning-driven Communication

Complex wireless communication systems managed by edge intelligence



Secure and private

Secure edge systems that ensure user privacy and keep information secure

Real-time training

New distributed algorithms that make it possible to build models almost in real-time

Nanophotonic technologies

Nanophotonic circuits will perform complex matrix operations

6G

6G Edge

First deployments of a new generation of Edge AI

2020 - 2030

Roadmap to edge intelligence

7

Edge computing is one of the key technologies that is enabling 5G networks to satisfy the stringent requirements in different use cases such as URLLC (contributing to low latency) or mMTC (providing distributed computing power). AI goes even beyond becoming essential techniques in the technology industry for implementing a wide variety of applications such as video processing, data analysis, image generation, and so on. Thus, as shown in the "Related work" section of this white paper, there is no doubt that both technologies will be combined in edge intelligence to play an important role in 6G.

The evolution to the deployment of a new generation of edge intelligence systems, applications, and services will occur during the next ten years, with the completion of different technological steps that will provide new devices, technology, and applications, as shown in the roadmap below.

Several challenges must be addressed. For example, hardware needs to evolve to make it economically viable to deploy (at the edge) a large number of efficient architectures and devices supporting existing and new AI techniques with high computing requirements, such as DNN, as well as supporting the larger amounts of data that will transit in future 6G wireless communications networks. Software will also advance in different aspects such as distribution, automation, intelligent orchestration of components, security, etc.

We thus anticipate steady sustained work that will address the main challenges identified in this white paper, providing new results in the various technological aspects related to edge intelligence. For example, the first deployments will use -trained models, but this will have to progress into systems that combine -trained and online learned models that will be able to define actions based on the information collected, even in real time. Despite the increasing power of edge hardware, it will be necessary to distribute both training and data processing. New hardware components (e.g. new AI accelerator application-specific integrated circuits) will be developed to improve performance, while reducing the energy consumed and costs. In the long term, technologies such as nanophotonics may be used to perform complex operations or store information. Communications and learning will be combined and exploited by learning-driven design principles. In addition, by exploiting the distributed nature of edge architecture, new proposals will arise to keep the data and intellectual property secure and ensure user privacy.

References

[1] Gagandeep Singh, Lorenzo Chelini, Stefano Corda, Ahsan Javed Awan, Sander Stuijk, Roel Jordans, Henk Corporaal, and Albert-Jan Boonstra, "Near-memory computing: Past, sent, and Future," November 2019, volume 71, 102868.

[2] Brian Gaide, Dinesh Gaitonde, Chirag Ravishankar, and Trevor Bauer. 2019. Xilinx Adaptive Compute Acceleration Platform: VersalTM Architecture. In Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '19). Association for Computing Machinery, New York, NY, USA, 84–93.

[3] Ali, Samad, et al. "6G White Paper on Machine Learning in Wireless Communication Networks," 6G Flagship, University of Oulu, June 2020, http://urn.fi/urn:isbn:9789526226736

[4] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," 2019. Proceedings of the IEEE.

[5] D. Xu, T. Li, Y. Li, X. Su, S. Tarkoma, P. and Hui, 2020. A Survey on Edge Intelligence. arXiv print arXiv:2003.12172

[6] Thomas Rausch and Schahram Dustdar, "Edge Intelligence: The Convergence of Humans, Things, and AI," June 2019, IEEE International Conference on Cloud Engineering (IC2E '19), Prague, Czech Republic.

[7] Xingzhou Zhang, Yifan Wang, Sidi Lu, Liangkai Liu, Lanyu Xu, and Weisong Shi, "OpenEI: An Open Framework for Edge Intelligence," Distributed Computing Systems (ICDCS) 2019 IEEE 39th International Conference on, pp. 1840–1851, 2019.

[8] Nitinder Mohan, "Edge Computing Platforms and Protocols," PhD Thesis. November 2019.

[9] Hamm, Andrea, Alexander Willner, and Ina Schieferdecker. "Edge Computing: A Comprehensive Survey of Current Initiatives and a Roadmap for a Sustainable Edge Computing Development." arXiv preprint arXiv:1912.08530 (2019).

[10] J. Park, S. Samarakoon, M. Bennis, and M. Debbah. "Wireless network intelligence at the edge," in Proceedings of the IEEE, vol. 107, no. 11, pp. 2204–2239, November 2019.

[11] Lovén, Lauri, et al. "EdgeAI: A Vision for Distributed, Edge-native Artificial Intelligence in Future 6G Networks," The 1st 6G Wireless Summit (2019): 1–2

[12] S. Deng, H. Zhao, J. Yin, and H. Dustdar. "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," 2020. IEEE Internet of Things Journal

[13] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Towards an intelligent edge: Wireless communication meets machine learning," -print arXiv:1809.00343, 2018.

[14] P. Hu, S. Dhelim, H. Ning, and T. Qiu (2017). "Survey on fog computing: architecture, key technologies, applications and open issues," Journal of Network and Computer Applications, 98, 27–42.

[15] Mahadev Satyanarayanan, Paramvir Bahl, Ramon Caceres, and Nigel Davies. "The Case for (VM)-based Cloudlets in Mobile Computing," 2009, IEEE Pervasive Computing, volume 8, issue 4, pages 14–23)

[16] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, et al. (2015). "Human-level control through deep reinforcement learning," Nature, 518(7540), 529–533.

[17] C. Savaglio, M. Ganzha, M. Paprzycki, C. Bădică, M. Ivanović, and G. Fortino (2020). Agent-based Internet of Things: State-of-the-art and research challenges. Future Generation Computer Systems, 102, 1038–1053.

[18] Leppänen, T. (2018). Resource-oriented mobile agent and software framework for the Internet of Things. Dissertation, Acta Universitatis Ouluensis, C Technica, (645).

[19] G. Fortino, W. Russo, C. Savaglio, W. Shen, and M. Zhou (2017). Agent-oriented cooperative smart objects: From IoT system design to implementation. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48(11), 1939–1956.

[20] T. Leppänen, J. Riekki, M. Liu, E. Harjula, and T. Ojala. "Mobile Agents-based Smart Objects for the Internet of Things," In: Fortino and Trunfio (eds.), Internet of Things based on Smart Objects: Technology, Middleware and Applications, pp. 29–48, ISBN 978-3-319-00491-4, Springer, 2014.

[21] Jihong Park, Sumudu Samarakoon, Hamid Shiri, Mohamed K. Abdel-Aziz, Takayuki Nishio, Anis Elgabli, and Mehdi Bennis, "Extreme URLLC: Vision, Challenges, and Key Enablers," Jan. 2020, arXiv:2001.09683.

[22] B. Zhang, A. Davoodi, and Y.H. Hu, "Exploring energy and accuracy tradeoff in structure simplification of trained deep neural networks," IEEE J. Emerg. Sel. Topics Circuits Syst., vol. 8, no. 4, pp. 836–848, December 2018.

[23] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, "Split learning for health: Distributed deep learning without sharing raw patient data," December 2018, arXiv:1812.00564.

[24] J. Konecný, H.B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," October 2016, arXiv:1610.02527.

[25] G.E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in Proc. NIPS Workshop Deep Learn., Montréal, QC, Canada, Dec.ember 2014,
[26] Peter Kairouz et. el. "Advances and Open Problems in Federated Learning," December 2019, arXiv:1912.04977.

[27] A. Elgabli, J. Park, A.S. Bedi, M. Bennis, and V. Aggarwal, "Q-GADMM: Quantized Group ADMM for Communication Efficient Decentralized Machine Learning," October 2019, arXiv:1910.10453

[28] N.H. Motlagh, E. Lagerspetz, P. Nurmi, X. Li, S. Varjonen, J. Mineraud, M. Siekkinen, A. Rebeiro-Hargrave, T. Hussein, T. Petaja, and M. Kulmala, 2020. "Toward massive scale air quality monitoring," IEEE Communications Magazine, 58(2), pp. 54–59.

[29] M. Ylianttila, R. Kantola, A. Gurtov, L. Mucchi, I. Oppermann (eds), "6G White paper: Research challenges for Trust, Security and Privacy". 6G Flagship, University of Oulu, June 2020, http://urn.fi/urn:isbn:9789526226804

6G White Paper on Edge Intelligence

Authors

Ella Peltonen, University of Oulu, Finland, ella.peltonen@oulu.fi · Mehdi Bennis, University of Oulu, Finland, mehdi.bennis@oulu.fi · Michele Capobianco, Capobianco, Italy, michele@capobianco.net · Merouane Debbah, Huawei, France, merouane.debbah@huawei.com · Aaron Ding, TU Delft, Netherlands, aaron.ding@tudelft.nl · Felipe Gil-Castiñeira, University of Vigo, Spain, xil@gti.uvigo.es · Marko Jurmu, VTT Technical Research Centre of Finland, Finland, marko.jurmu@vtt.fi · Teemu Karvonen, University of Oulu, Finland, teemu.3.karvonen@oulu.fi · Markus Kelanti, University of Oulu, Finland, markus.kelanti@oulu.fi · Adrian Kliks, Poznan University of Technology, Poland, adrian.kliks@put.poznan.pl · Teemu Leppänen, University of Oulu, Finland, teemu.leppanen@oulu.fi · Lauri Lovén, University of Oulu, Finland, lauri.loven@oulu.fi · Tommi Mikkonen, University of Helsinki, Finland, tommi.mikkonen@helsinki.fi · Ashwin Rao, University of Helsinki, Finland, ashwin.rao@helsinki.fi · Sumudu Samarakoon, University of Oulu, Finland, sumudu.samarakoon@oulu.fi · Kari Seppänen, VTT Technical Research Centre of Finland, Finland, kari.seppanen@vtt.fi · Paweł Sroka, Poznan University of Technology, Poland, pawel.sroka@put.poznan.pl · Sasu Tarkoma, University of Helsinki, Finland, sasu.tarkoma@helsinki.fi · Tingting Yang, Pengcheng Laboratory, China, yangtt@pcl.ac.cn

6G Flagship, University of Oulu, Finland
June 2020

6G Research Visions, No. 8
ISSN 2669-9621 (print)
ISSN 2669-963X (online)
ISBN 978-952-62-2677-4 (online)

6G FLAGSHIP
UNIVERSITY OF OULU

6gflagship.com