7

Intelligent Management at the Edge

Mohammadreza Mosahebfard[1], Claudia Torres-Pérez[1], Estela Carmona-Cejudo[1], Andrés Cárdenas Córdova[1], Adrián Pino Martínez[1], Juan Sebastian Camargo Barragan[1], Estefanía Coronado[1,2], and Muhammad Shuaib Siddiqui[1]

[1]i2CAT Foundation, Spain
[2]Universidad de Castilla-La Mancha, Spain
E-mail: reza.mosahebfard; claudia.torres; estela.carmona; andres.cardenas; adrian.pino; juan.camargo; estefania.coronado; shuaib.siddiqui@i2cat.net; estefania.coronado@uclm.es

Abstract

AI/ML techniques play a key role in 5G/6G networks providing connectivity to IoT devices. In such scenarios, not only is it necessary to run time-sensitive applications with strict latency requirements without human intervention, but it is also key to apply automation techniques at both the application and the network levels. The chapter is composed of three sections. In the first section, we present different cloud native (CN) technologies enabling scalable, cost-efficient, and reliable IoT solutions. The second section details different distributed and hierarchical monitoring frameworks and metrics collection schemes as inputs to AI engines. In the last section, application placement problems focused on delay minimization in geographically distributed single-cluster environments are first discussed. Afterwards, application placement issues ensuring latency requirements for the applications and energy consumption in distributed multi-access edge computing (MEC) systems using AI pipelines are presented.

Keywords: AI/ML, edge computing, edge intelligence, edge optimization, edge automation, 5G/6G networks, IoT, monitoring frameworks, distributed MEC, application placement.

7.1 Introduction to Intelligence at 5G/6G Networks Edge

Edge computing refers to bringing computing resources and capabilities closer to the devices that generate or consume data. This can help to reduce latency, improve performance, and increase security. It also facilitates both edge automation and intelligence. On the other hand, according to 5GPPP, high-performance next generation networks will be operated via a scalable management framework enabling service provisioning time from 90 hours to 90 minutes, by reducing the network management OPEX by at least 20% compared to current networks [1]. A promising solution to achieve 5G networks with a level of intelligence similar to that of humans as well as lower levels of latency is the combination of artificial intelligence (AI) and edge computing. AI at the edge refers to the use of AI algorithms and models at the edge of a network, closer to the end-user generating or consuming the data, which results in performance improvement and latency reduction.

7.1.1 Edge automation

7.1.1.1 State of the art

Two of the main international organizations and standardization bodies, namely 3GPP and ETSI, have defined requirements, features, and key technologies in the context of the 5G edge. The 5G 3GPP system architecture [2] is intended to support edge computing by enabling services such as the Internet of Things (IoT), industrial solutions, smart energy, connected health, autonomous driving and more. Another contribution from 3GPP involves studying the management aspects of edge computing, where several edge scenarios and use cases are explored and potential deployment solutions are discussed [3]. Following this line of work, enhancements regarding edge computing management and connectivity models have been proposed [4], which include a number of concepts such as self-organizing networks (SON) and network data analytics function (NWDAF). SON is an automation technology designed to streamline and simplify planning, configuration, management, optimization, and healing. SON architectures are conceived in three variants, centralized SON, distributed SON, and hybrid SON. Each variant is a key technology with the main aim of integrating legacy mobile radio access networks (RAN) [5]. Recent advancements in AI/ML techniques have led to an increased interest in SON with cognitive features combined with the software/hardware decoupling movement – via network function virtualization (NFV), and/or multi-access edge computing (MEC) – leading

to greater network agility. NWDAF was introduced to provide a standard method to collect data supporting 5G core network functions and operation administrations and management systems [6].

ETSI has also published several reference architectures and specifications of the aforementioned NFV and MEC initiatives. By using zero-touch network and service management (ZSM), end-to-end network management can be achieved with minimal or no human intervention. ZSM facilitates collaborative management interactions between all layers of the network through the use of closed-loop automation, AI, adaptive ML, and cognitive technologies [7], abstracting the 5G network edge resource management. On the radio side, open RAN refers to the disaggregation movement of hardware and software in wireless telecommunications as well as to create open interfaces between them [8].

7.1.1.2 Key enablers

To meet edge automation expectations several vital technologies are required, including distributed data collection, real-time processing, and edge automation for 5G slicing. Both distributed data collection and real-time processing require streaming, in-memory storage management, and computing close to the edge in order to minimize latency and maximize bandwidth. In addition, stakeholders need to plan, design, and activate several customized network slices rapidly to provide customers with different 5G services. Slice elasticity, the ability to scale up or down in response to performance changes, also has become a must. To this end, by forecasting the upcoming traffic with AI/ML techniques, network slices can be optimized by minimizing resource usage while meeting quality of service (QoS) or customer requirements. A critical component of successful 5G service delivery is network slicing. A network slice is considered as a collection of networking and computational resources forming a dedicated network that provides an end-to-end connectivity to hosted applications and services [9]. Stakeholders are able to plan, design, and activate several customized network slices on demand. Moreover, slice elasticity, which is defined as the ability to scale up or down in response to variations in performance, is critical. In this regard, AI/ML techniques play an important role, since forecasting the upcoming traffic allows the slice to be adjusted (using a proactive rather than reactive model) to minimize resource consumption, meet QoS requirements, and perform lifecycle management tasks on existing slices.

7.1.2 Edge intelligence

5G/6G networks and AI/ML are closely related with edge devices of limited computing power are able to leverage 5G/6G network edge intelligence by distributing the computation, which is driven by the use of AI/ML techniques and distributed intelligence. A joint perception environment could be formed of real-time metrics collected from devices in the network. A perception environment of this type groups decisions in order to enhance the efficiency, productivity, and safety of several 5G edge applications. Such shared intelligence will be enhanced by the use of a hybrid and distributed architecture. By combining 5G edge networks with MEC architectures, distributed learning [10], and collaborative intelligence [11], real-time distributed intelligence and collaboration are becoming tangible. Intent-based networking [12], which has recently been applied to the RAN, is another promising idea that is undergoing development and adaptation for B5G networks.

7.1.2.1 State of the art

A flexible and hybrid architecture, both centralized and distributed, is critical for edge intelligence architectures. In terms of communication, a number of developments have been made, including direct device-to-device and multi-hop communication, which are mentioned in 3GPP standards [13]. They have been combined with 5G scenarios via the cellular vehicle-to-everything (V2X) paradigm to meet KPIs in verticals such as autonomous driving. In terms of radio management, intent-based RAN management is becoming increasingly important. It consists of altering the configuration of the RAN from the setting of technical parameters to the specification of connectivity services, allowing service providers to prioritize users and services based on their device capabilities and use cases.

Another integral part of edge intelligence is real-time access and analysis of data, along with concepts such as explainable AI (XAI), named data networks, joint optimization of communication and computing, distributed machine learning, and meta-learning, which are examples of technologies that will pave the way for B5G and 6G edge networks [14].

7.1.2.2 Key enablers

XAI is a set of methods and techniques for producing accurate and explainable models, along with explaining how and why the algorithm arrives at a specific solution, leading to an output that is comprehensible and transparent for humans. Another technology that is helping to meet the increasingly

ambitious performance requirements is multi-access traffic management at the edge. By using the multi-access protocol and multiple access management [15], different technologies can be handled seamlessly. A multi-access protocol stack consists of two layers; a convergence sublayer that manages access path selection, multi-link aggregation, and more multi-access-specific tasks, and an adaptation sublayer that handles tunneling, security, and NAT.

In addition, joint optimization of computation and communication is quite a transcendental point to take into account in 5G/6G networks, as it helps to improve performance while managing both computation and radio resources intelligently. Lastly, distributed and federated learning are techniques that enable edge intelligence without transferring data to the cloud. Such learning techniques employ a collaborative learning model in which each element has a partial view of the system. As opposed to fully distributed learning where nodes must collaborate peer-to-peer, federated learning manages the collaboration through a central coordinator.

7.1.3 Edge computing and 5G/6G: a cloud native architecture

The current edge computing ecosystem is dynamic and evolutionary, which is the combination of the classic edge computing with several existing technologies and techniques including cellular networks, CN, and AI/ML. Thus, there is no de facto standard set of tools for implementing 5G/B5G edge computing architectures; however, the direction of such edges is becoming clearer. A number of factors have been identified as driving the adoption and evolution of edge architectures [16]. These include connectivity, applications exposed via APIs, the use of increasingly intelligent orchestrators, service exposure and optimization, and free open-source software [17].

From a technological point of view, CN technologies seem to be a perfect fit for edge architectures. In order to meet emerging 5G standards and provide flexibility for multi-vendor managed networks, edge solutions that are based on automation and intelligence need to be designed and developed as cloud-native architecture. The concept of CN is to decompose applications into a set of microservices that can be developed and deployed independently, in order to accelerate and optimize the DevOps lifecycle of software systems. A container orchestrator is responsible to schedule microservices to run on compute nodes by packaging them into lightweight containers. The CN approach is concerned with the way applications are developed and deployed, rather than only the place where they are executed [18]. Kubernetes, also known as k8s, has been adopted by the Cloud Native Computing Foundation

(CNCF) [19] as the open-source management tool for microservice-oriented applications. In CN architectures, streaming solutions such as Kafka [20] and Rabbit-MQ [21] are seamlessly integrated, along with publish−subscribe protocols such as MQTT [22] and data lake technologies such as Spark, which, by generating insights on edge nodes, reduces the need to transport data all the way to the cloud. In spite of the fact that these technologies were developed for different requirements, they complement each other perfectly in certain circumstances.

Container technology and Kubernetes orchestration framework provide scalability, cost-efficient, and reliable solutions. Hybrid k8s clusters with heterogeneous architectures provide the flexibility needed for the successful implementation of IoT applications. As the number of microservices in a scenario increases, it can be challenging to understand the interactions and identify and track errors. The service meshes can be used to resolve this problem [23], where linkerd [24] are currently being positioned as the *de facto* solution to the problem. Due to the operator's trend, Kubernetes has evolved from a declarative to an imperative model, where a set of controllers perform the required actions to match the intended state. OpenShift [25] is an example of a tool that adopts this concept, while several aspects, such as multi-cluster management, multi-cloud connectivity solutions, and workload migration, require further investigation.

Furthermore, 5G/6G edge architectures could benefit from the adoption of extended Berkeley filter packer (eBPF) technology [26]. It is emerging that different tools based on this technology, such as Cilium [27], allow a code to run within the kernel without the need to compile the entire kernel, providing unparalleled flexibility, as well as promising improvements in key areas such as security, networking, and monitoring, where AI will have a significant impact.

7.2 Distributed Telemetry

The field of intelligent networking has gained momentum in recent years due to the popularity of machine learning models and artificial intelligence systems in the telecommunications industry [28]. The concept of intelligent networking is mainly concerned with optimizing the management and performance of different network segments, such as radio, computing, and transport networks, each of which has heterogeneous objectives and approaches. As an example, some concepts, such as SON, address autonomic or cognitive

self-managed networks [29]. Nevertheless, to cope with those characteristics, cognitive self-managed systems require strong telemetry systems to be aware of the behavior and performance of each of the elements composing the network infrastructure and the service communications. It is the consistent metrics that feed into the self-management systems enabling intelligent management models to achieve better results and, therefore, improve the performance of communication networks. However, due to the nature of current networks, thons of metrics gathered from segments that span several administrative domains significantly increase the complexity of the telemetry systems. This means that telemetry systems should be able to provide well-organized and differentiated metrics from each source so that they may be able to expose metrics per customer, per service, and per network element on demand.

As 5G networks are based on cloud-native and distributed services, multiple logical networks can be created and coexisted in a common infrastructure through technological enablers such as NFV [30], software defined networking (SDN) [31], and edge/cloud computing. Logical networks refer to the network slicing communication paradigm enabled by 5G networks by nature, which allows for the allocation of slices per service and per client. Since network slicing spans different network segments, edge computing must be capable of dealing with network slicing capabilities [32]. To meet the performance requirements and quality of service expected by users, several critical, time-sensitive, and less-consuming services are being moved to edge computing [33]. As a result, intelligent systems are also moving toward edge environments so that they can manage different services running at the edge that may belong to different vertical clients or network slices. Telemetry systems must adapt to paradigms such as network slicing, multi-tenancy, and multi-domain as well as to environments so that they can monitor aspects of these services in a flexible and dynamic manner. Monitoring systems may have to update their sources where metrics are collected frequently when services change.

Basically, the telemetry systems are a control framework that gives a detailed view of the state of a system. It allows assuring the desired operation of infrastructure resources as well as to analyze the performance of each virtualized service. The monitoring systems have existed since the emergence of IP networks with the aim to mitigate failures, attacks, and undesired behavior. As networks have evolved, monitoring systems have adapted and sophisticated their metrics acquisition models to better address unpredictable (proactive) and predictable (reactive) situations that violate

operator-provided service level agreements (SLAs). Addressing proactively a monitoring situation means foreseeing events that can be mitigated in advance through the execution of specific actions. Proactive methods are based entirely on machine learning models that analyze patterns in historical data and anticipate future behavior. This is the core concept where intelligent networks are built. However, reactive methods refer to executing actions at the exact moment that an event occurs, which violates the SLAs. This principle has been widely used in most control systems. However, since the democratization of machine learning models, control systems are tending to use hybrid control methods depending on the requirements of SLAs and use cases. However, the performance methods are independent of the monitoring systems but depend on the type and quality of metrics they receive from the monitoring systems. Consequently, monitoring systems must meet the needs of each method to assure adequate control of services and resource infrastructure. In terms of monitoring system design, it is difficult to anticipate all the needs of the methods, but if they provide better visibility of each of the elements that comprise the communication service, the methods will be more likely to provide better performance.

In this context, previous research has focused on specific aspects of monitoring. For example, in [34], the authors make a study on traffic differentiation detection where they focus on presenting strategies and tools to monitor network traffic. On the other hand, in [35], the authors present a survey on network security monitoring. Here, the paper reviews the approaches and tools focused on network security aspects. In [36], the authors focus their attention on an exhaustive study of platforms for monitoring cloud environments. They detail both licensed and open-source tools. The important aspect of a monitoring system is to be able to perform all these types of monitoring with a single robust telemetry framework.

The following sections will provide a detailed description of the hierarchical and distributed monitoring architectural framework for 5G and 6G networks that provide flexibility and visibility of metrics obtained from both communication services and network infrastructure. Section 7.2.1 gives a detailed description of each component composing the architectural framework.

7.2.1 Hierarchical and distributed monitoring framework

The main objective of the distributed and hierarchical monitoring framework is to collect, organize, and expose the data flow, resource, and configuration

metrics generated by each of the network segments. The system is hierarchical because its components are distributed across several layers of view or management levels where data is aggregated, filtered, and isolated. This allows metrics to be persisted and exposed at different levels, even with different levels of granularity. The different levels of monitoring are fed by separate and distributed monitoring agents deployed by the operator in each network segment.

Figure 7.1 illustrates the design of the architectural framework of the hierarchical and distributed monitoring system. In this case, two levels of metrics abstraction are defined. In addition, each level allows centralizing and persisting the metrics obtained from the network segments. This makes it easier for each network segment to have several monitoring agents and a common metrics centralizer. For example, for access networks such as Wi-Fi, small cells, and eNBs, monitoring agents could be deployed for each of them to interact directly and to extract the metrics generated in each network equipment. These monitoring agents are then aggregated to the first-level aggregators, where the metrics can be exposed and visualized by customers and operators. The same case would be for NFV infrastructure (NFVI) nodes, where there will be several types of monitoring agents deployed, both for the NFV node itself and for each of the virtualized network functions (VNFs) running on it. Similarly, these metrics may be aggregated, exposed, and visualized by one or more top-level aggregators, depending on the need of the use cases or customers. However, the communication service and network infrastructure of a network operator may be composed of multiple access networks, NFVI nodes, and transport networks; so there will be multiple first level aggregators. This is the motivation behind the use of a second level of aggregation, where the metrics collected by the first level aggregators are centralized. The second level of aggregation allows a network operator and customers in general to have a global view of the current state of the network infrastructure and the communication services running on it. It facilitates filtering by first-level aggregation nodes, without having to worry about which monitoring agent is being referred to when extracting a metric.

7.2.1.1 Monitoring agents

Monitoring agents are software tools that interact directly with network elements. They can be run directly on the network equipment or they can be run as services in edge/cloud computing. Monitoring agents are known as node exporters, which take all the metrics and push them to the top-level aggregator

so that they can be understood and visualized. There are monitoring agents designed by default for different types of network elements, while others can be customized (pushgateways) and run as a set of scripts that interact directly with the operating system of the network element to extract the metric.

7.2.1.2 Aggregators – monitoring servers

Aggregators are instances of time series databases (TSDB) in charge of collecting and centralizing the metrics exposed by the monitoring agents. The aggregators persist the metrics for a given time to allow operators, users, or other components to access the historical information provided by the monitoring agents. In addition, they allow metrics to be visualized and operationalized to contextualize them in human-understandable units of measurement. Currently, many of the network services are deployed in conjunction with a metrics aggregator dynamically, which generates the need to implement a static second-level aggregator. There are several alternatives

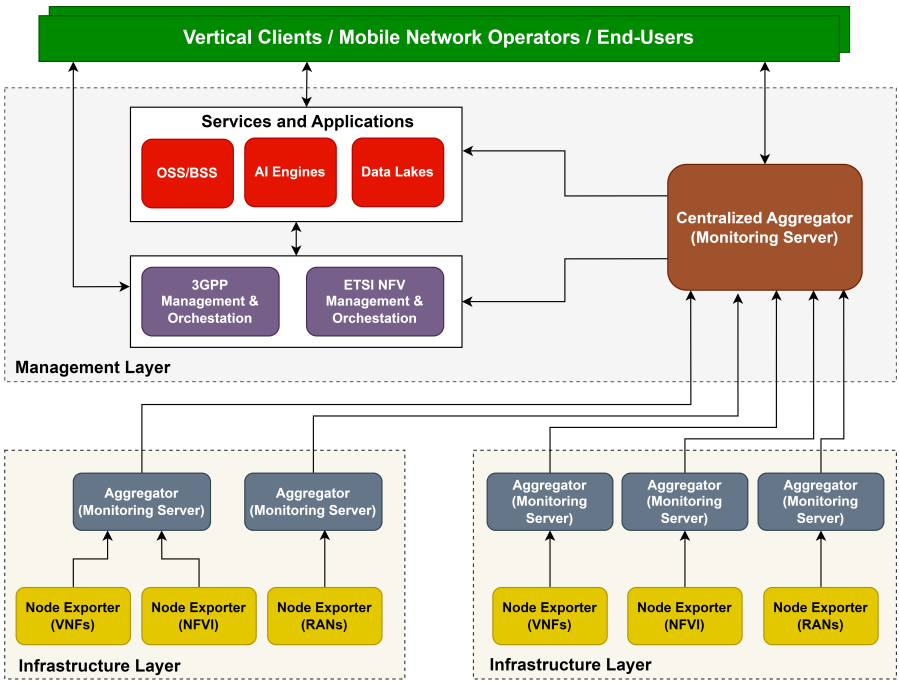

Figure 7.1 Architectural framework of the distributed and hierarchical monitoring system.

in the TSDB market; however, the most popular ones are Prometheus[1], InfluxDB[2], TimeStream[3], and TimescaleDB[4]

7.2.1.3 Centralized aggregator – monitoring server

The centralized aggregator is in charge of collecting the metrics exposed by the first-level aggregators. In other words, it adds the first-level aggregators as direct targets and is not aware of the number of monitoring agents that exist in the system. This level of abstraction allows operators to dynamically scale and manage first-level aggregators that are dynamically deployed alongside network services. On the other hand, the centralized aggregator also allows visualizing the metrics exposed by all monitoring agents by filtering them by each first-level aggregator. One tool that acts as a centralized aggregator is Thanos[5]. It has the same working principle as Prometheus.

7.3 AI Pipelines for the Edge-to-cloud Continuum

While the development and deployment of 5G mobile networks is ongoing, extensive research efforts are currently being directed toward the requirements of future 6G mobile networks, covering aspects such as architecture, enabling technologies, key features, and requirements. Among these, network cloudification is one clear 6G architectural trend. Moreover, 5G network developments are already paving the way to support a massive number of end devices across the cloud continuum [37].

Research challenges related to the massification of end devices in 5G networks are often related to the placement of applications and network functions that might be distributed across multiple devices spanning the cloud continuum [38], and to the optimization of strict latency, reliability, and bandwidth requirements.

As the 6G paradigm introduces a shift to the full digitalization of the real world, some additional critical aspects need to be considered, such as efficient interworking with IoT devices, the support of advanced, novel edge computing solutions, and adequate cloud support for network operation. In this regard, the native support of AI and ML in 6G can provide innovative

[1] https://prometheus.io/
[2] https://www.influxdata.com/
[3] https://aws.amazon.com/es/timestream/
[4] https://www.timescale.com/
[5] https://thanos.io/

solutions, for example, related to the optimization of network functions and distributed applications [39]. AI and ML techniques will become critical to automate decision-making processes in 6G and enable the implementation of predictive orchestration mechanisms.

However, the intertwining of communication and computation algorithms in 6G requires suitable in-network governance mechanisms. In particular, every infrastructure and service component in the network must be controllable by the tenant, which requires very versatile, pervasive, and automatic resource control capabilities [40]. This calls for the design of a 6G-native AI fabric that caters for the diversity of resources and end devices across the cloud continuum, which should be able to provide not only novel, natively embedded governance capabilities but also the ability to optimize the use of resources in the network in an energy-efficient manner.

7.3.1 Native AI for distributed edge-to-cloud environments

6G is promising to become a networking technology whose management and behavior are meant to be closer to human's brain reasoning. The vision must also include the native incorporation of AI processes capable of handling network functions more efficiently (e.g., intelligent network management and wireless resource configuration) as well as training and executing AI-based models [41], [42].

Networking ecosystems have also evolved from the point of view of the distribution of the radio and computational resources. In this regard, future mobile networks are expected to be fully geographically distributed and managed by different entities and operators, and even based on several administrative domains (see Figure 7.2). Related to this, the highly distributed telemetry systems at different network segments make available huge data volumes which, although provide a full vision of the system's status, also multiply the difficulty in knowledge extraction. Therefore, despite the improvement expected in availability level and network performance, together with the high-dimensional data, it will greatly increase the complexity of management and error handling, making it impractical for human operators [43]. For that reason, an AI-enabled architecture able to build knowledge natively and act autonomously is the goal of 6G networks.

Adopting the aforementioned AI processes as well as regular user applications at the edge of the network brings, however, new challenges to next-generation networks. Undoubtedly, the increase in heterogeneity of both edge nodes and application requirements, the computational limitation of

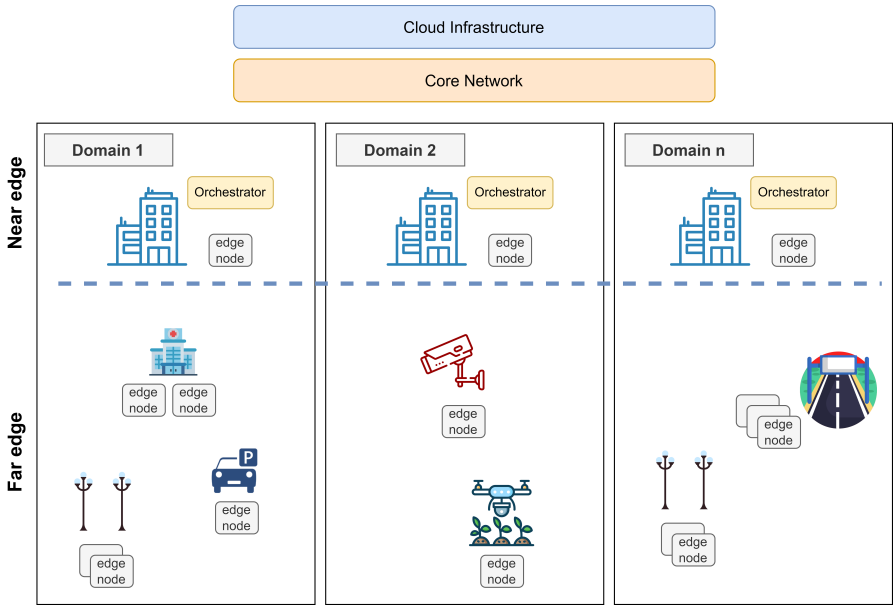

Figure 7.2 Example of a highly distributed and independently managed edge infrastructure.

the edge nodes, and the dynamic change of user demands make intelligent resource management approaches able to ensure the data privacy become essential [44]. More specifically, application and function placement can be considered one of the key resource allocation problems, especially as we deal with highly heterogeneous and distributed infrastructure involving computational and communication resources [45], [46]. Therefore, there is a need for intelligent and distributed placement solutions that provide decisions without sharing the data belonging to each administrative domain or independent system.

In this regard, distributed and federated learning have been demonstrated to provide excellent performance due to the ability to collaboratively build a model without data transferring, therefore avoiding data privacy issues and extra overheads in the data transmission process [47], [48]. Similarly, reinforcement learning has shown promising results in tackling this challenge in centralized scenarios, such as in the works proposed in [49] and [50].

Most of the recent research related to application placement is related to either (i) computational offloading at the edge from end-user devices, (ii) latency-aware processes at the radio side, and (iii) edge infrastructures where telemetry data is not distributed, or in which the various nodes are managed

by the same orchestration entity. On the one hand, offloading approaches for energy saving in the mobile devices tend to neglect the energy consumption of the edge servers, which are also more resource-constrained than cloud infrastructures. This issue could be made worse by uneven distributions of users in the geography, which could also make edge placement algorithms waste energy having nodes with very low resource utilization instead of being powered off. On the other hand, the maximum latency supported by applications must also consider the link delay depending on the placing node and the processing time. In essence, it should ensure that besides meeting the application requirements, also the QoS constraints are ensured in a unified manner, especially for the time-sensitive applications. In the next subsections, these problems are greatly discussed, especially when they are addressed by AI processes in highly distributed (and administratively independent) systems.

7.3.1.1 Energy saving in distributed edge computing

Extensive research has been performed in MEC to optimize the energy consumption of computationally intensive tasks, given the limited resources of the servers used. Application placement algorithms are increasingly important at the edge since, among other consequences, computational tasks offloaded to the cloud can result in lower utilization of MEC resources and higher power consumption. The performance of applications could be affected due to the demanding application requirements that can limit the storage and capacity of end devices. In addition, in future 6G networks, expected to be extremely geo-distributed in terms of computational resources, centralized orchestration approaches could lead to constant interaction between central entities and result in energy consumption.

The state of the art highlights the need to focus on the placement of applications and workloads that produce lower energy consumption. Moreover, it is to be considered not only the energy consumed by the application itself when it is running but also some transactions when moving applications across several nodes. This can be the case of the follow-me scenario. In this case, energy consumption on edge servers, migrations from edge servers to cloud servers and between edge servers must be taken into account. In addition, other approaches suggest maintaining the edge servers in an idle state or low consumption and activating the server when a new application arrives. However, not all works consider all possible sources of energy consumption, because depending on the use case, it might be more necessary to prioritize

the minimization of expenditure in some sources of consumption than in others.

Numerous research contributions that attempt to solve this problem aim to strike a balance between performance metrics and energy efficiency. Machine learning techniques have been widely used in this topic, due to their ability to make predictions from data and to obtain assumptions about the environment without prior knowledge. For application placement, forecasting methods predict periodic changes from time series considering the edge node data as input and the geographic location information [51]. The authors of [52] aim to reduce the total energy of each user, including local computation and wireless transmission energy under a federated learning approach. However, the energy consumption on only the terminal side is addressed in [53]. Reinforcement learning and its variants are oriented to minimize the long-term energy consumption and have been demonstrated to be a good alternative for these kinds of scenarios [54]. For instance, some authors consider application placement with multiple metrics in dynamic environments as a problem to solve with distributed learning approach [53].

7.3.1.2 Latency-aware AI processes in edge computing

As stated previously, one of the key enablers of the incoming generation of network services is the ability to bring the processing power near to the final user, using edge computing as a tool to decrease the potential delays in end-to-end communications. The management of this delay is particularly important in ultra-reliable low-latency communications (URLLC) as an inappropriate delay would generate misbehavior in time-sensitive applications, affecting use cases as diverse as smart living, Industry 4.0, or autonomous vehicles [55]. Essentially, selecting the proper host to implement the service application placement is critical if the stringy delay requirements of the applications are to be fulfilled. Contrary to what might be expected, the host's selection is not a trivial labor, as different elements contribute to the final decision. However, it is not sufficient to consider the current delay of the proposed hosts. Additionally, it is essential to account for the processing delay after the application has been instantiated in the server, the computational characteristics of the host, the distance between the host and the users, and an increasing number of secondary parameters.

Considering the previously mentioned constraints, human decision-making would be time-consuming and error-prone, making it necessary to implement an automated decision-making system instead. Traditional optimization models include the use of algorithms that perform numerical

analysis and mathematical optimization methods [56], [57]. However, considering the dynamicity of the network, a system that is able to adapt to this type of changes is necessary, excluding the possibility of using traditional optimization models. Incidentally, machine learning models excel in this type of conditions and are natively suited to handle data in time-series format and with an abundance of data categories. Machine learning models can solve optimization problems successfully and accurately and at the same time being flexible enough to adapt to the unique changes of the network, showing more generalization capabilities than its traditional counterpart.

As such, ML processes have been proven to be suitable for solving the best placement location for delay-constrained applications. When deployed on a centralized point of the network architecture, ML models use as input the parameters that are monitored through the network orchestrator or the network management service. These parameters are affected directly or indirectly by the end-to-end delay; so it is especially important to measure KPIs that are linked with the propagation delay, the processing delay, and the radio communication delays, among others. Under this statement, the authors of [58] look to maximize the quality of experience (QoE) by analyzing packet loss rate, packet error rate, and latency under a two-level deep reinforcement learning model that suggests the best application position. Similarly, in [59], a deep reinforcement learning model is introduced, which uses transmission delay, propagation delay, and execution delay to reach a compromise between the application requirements and the server capacity. Finally, the work in [60] uses parameters directly obtained from the end-users, in a deep reinforcement learning configuration, to generate a tradeoff between the current performance delay-oriented and the cost of running the application. To do so, it searches for a balance between the delay experienced by the user and the cost taken from the network provider while distributing the application. Consequently, according to the state of the art, deep reinforcement learning is a good fit for scenarios whose initial inputs are unknown and adapts well to the latency-related metrics in application placement problems, providing flexibility and adaptability to an ever-changing network environment.

Acknowledgements

This work has been mainly supported by the EU H2020 research and innovation program IoT-NGIN with Grant Agreement No. 957246. It has also been supported by the EU "NextGenerationEU/PRTR," MCIN, and

AEI (Spain) under project IJC2020-043058-I, and by the Grant ONOFRE-3 PID2020-112675RB-C43 funded by MCIN/AEI/10.13039/501100011033.

References

[1] 5GPPP, A Pre-Structuring Proposal Based on the H2020 Work Programme. [Online]. Available: https://5g-ppp.eu/coverage-plans/

[2] 3GPP, System Architecture for the 5G System (5GS), V.17.1.1, 3GPP TS 23.501, 2021.

[3] 3GPP, Study on management aspects of edge computing, V16.0.1, 3GPP TR 28.803, 2019.

[4] 3GPP, 5G System Enhancements for Edge Computing, V1.0.0,, 3GPP TS 23.54, 2021.

[5] 5G; Self-Organizing Networks (SON) for 5G networks (3GPP TS 28.313 version 16.0.0 Release 16), 2020.

[6] 3GPP, 5G; 5G System; Network Data Analytics Services; Stage 3 (3GPP TS 29.520 version 15.3.0 Release 15), 2019.

[7] ETSI, "Zero-touch network and Service Management (ZSM); Reference Architecture, ETSI GS ZSM 002 V1.1.1," 2019.

[8] O-RAN, "O-RAN Architecture Description", v05.00, O-RAN, WG1, 2021.

[9] A. Papageorgiouet al., "On 5G network slice modelling: Service-, resource-, or deployment-driven?" *Computer Communications*, vol. 149, pp. 232–240, 2020, doi: 10.1016/j.comcom.2019.10.024.

[10] P. S. Dutta, N. R. Jennings and L. Moreau, "Cooperative Information Sharing to Improve Distributed Learning in Multi-Agent Systems," *Journal of Artificial Intelligence Research*, vol. 24, p. 407–463, 2005, doi: 10.1613/jair.1735.

[11] I. V. Bajić, W. Lin and Y. Tian, "Collaborative Intelligence: Challenges and Opportunities,"*2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8493-8497, 2021, doi: 10.1109/ICASSP39728.2021.9413943.

[12] Cisco, "Intent based networking". [Online]. Available: https://www.cisco.com/c/en/us/solutions/intent-based-networking.html

[13] 3GPP, "Overall Description of Radio Access Network (RAN) Aspects for Vehicle-to-Everything (V2X) based on LTE and NR", 3GPP TR 37.985, V.16.0.0," 2020.

[14] 3GPP, "Integrated Access and Backhaul Radio Transmission and Reception", 3GPP TS 38.174, V.16.3.0," 2021.
[15] 5G Americas, "5G Edge Automation and Intelligence" White Paper, 2021.
[16] RFC 8743, "Multi-Access Management Service," [Online]. Available: https://www.rfc-editor.org/rfc/rfc8743.txt
[17] Ericsson, "Edge computing and deployment strategies for communication service providers," [Online]. Available: https://www.ericsson.com/en/reports-and-papers/white-papers/edge-computing-and-deployment-strategies-for-communication-service-providers.
[18] 5G Americas, "Distributed Compute and Communications in 5G" White Paper, 2022.
[19] Cloud Native Computing Foundation (CNCF). CNCF Cloud Native Definition v1.0, [Online]. Available: https://github.com/cncf/toc/blob/master/definition.md
[20] Kafka. [online]. Available: https://kafka.apache.org/
[21] RabbitMQ. [online]. Available: https://www.rabbitmq.com/
[22] MQTT. [online]. Available: https://mqtt.org/
[23] Linkerd Documentation. What is a service mesh? [online]. Available: https://linkerd.io/what-is-a-service-mesh/
[24] Linkerd. [Online]. Available: https://linkerd.io/
[25] OpenShift. [Online]. Available: https://www.redhat.com/en/technologies/cloud-computing/openshift
[26] eBPF Documentation. What is eBPF? [online]. Available: https://ebpf.io/what-is-ebpf
[27] Cilium. [Online]. Available: https://cilium.io/
[28] P. V. Klaine et al., "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," in *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2392-2431, 2017, doi: 10.1109/COMST.2017.2727878.
[29] T. Meriem et al., "ETSI white paper no. 16 gana - generic autonomic networking architecture reference model for autonomic networking, cognitive networking and self-management of networks and services," 2017.
[30] M. Mosahebfard, J. S. Vardakas and C. Verikoukis, "Modelling the Admission Ratio in NFV-Based Converged Optical-Wireless 5G Networks," *in IEEE Transactions on Vehicular Technology*, vol. 70, no. 11, pp. 12024-12038, 2021, doi: 10.1109/TVT.2021.3113838.

[31] M. Dalgitsis et al., "SDN-Based Resource Management for Optical-Wireless Fronthaul," in: *(eds) Enabling 6G Mobile Networks*, Springer, Cham. 2022, doi: 10.1007/978-3-030-74648-3_14.

[32] A. Cárdenas et al., "Enhancing a 5G Network Slicing Management Model to Improve the Support of Mobile Virtual Network Operators," in *IEEE Access*, vol. 9, pp. 131382-131399, 2021, doi: 10.1109/ACCESS.2021.3114645.

[33] J.-M. Fernandez, I. Vidal, and F. Valera, "Enabling the orchestration of IoT slices through edge and cloud microservice platforms," *Sensors*, vol. 19, no. 13, p. 2980, 2019, doi: 10.3390/s19132980.

[34] H.-C. Hsieh, J.-L. Chen, and A. Benslimane, "5G virtualized multi-access edge computing platform for IoT applications," *Journal of Network and Computer Applications*, vol. 115, pp. 94–102, 2018, doi: 10.1016/j.jnca.2018.05.001.

[35] T. Garrett et al., "Monitoring Network Neutrality: A Survey on Traffic Differentiation Detection," in *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2486-2517, 2018, doi: 10.1109/COMST.2018.2812641.

[36] I. Ghafir et al., "A Survey on Network Security Monitoring Systems," *IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, pp. 77-82, 2016, doi: 10.1109/W-FiCloud.2016.30.

[37] G. Aceto et al., "Cloud monitoring: A survey," *Computer Networks*, vol. 57, no. 9, pp. 2093-2115, 2013, doi: 10.1016/j.comnet.2013.04.001.

[38] The 5G Infrastructure Association (5GIA), "European Vision for the 6G Network Ecosystem", white paper, 2021, doi: 10.5281/zenodo.5007671.

[39] E. Carmona Cejudo, and M. S. Siddiqui, "An Optimization Framework for Edge-to-Cloud Offloading of Kubernetes Pods in V2X Scenarios," *IEEE Globecom Workshops (GC Wkshps)*, 2021, doi: 10.1109/GCWkshps52748.2021.9682148.

[40] M. Ericson et al., "6G Architectural Trends and Enablers," *IEEE 4th 5G World Forum (5GWF)*, pp. 406-411, 2021, doi: 10.1109/5GWF52925.2021.00078.

[41] K. B. Letaief et al., "Edge Artificial Intelligence for 6G: Vision, Enabling Technologies, and Applications," in *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 5-36, 2022, doi: 10.1109/JSAC.2021.3126076.

[42] A. Bandi, "A Review Towards AI Empowered 6G Communication Requirements, Applications, and Technologies in Mobile

Edge Computing," *6th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 12-17, 2022 doi: 10.1109/ICCMC53470.2022.9754049.

[43] E. Coronado et al., "Zero Touch Management: A Survey of Network Automation Solutions for 5G and 6G Networks," in *IEEE Communications Surveys & Tutorials*, 2022, doi: 10.1109/COMST.2022.3212586.

[44] M. Giordani et al., "Toward 6G networks: Use cases and technologies", *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55-61, 2020, doi: 10.1109/MCOM.001.1900411.

[45] C. R. de Mendoza et al., "Near Optimal VNF Placement in Edge-Enabled 6G Networks," *25th Conference on Innovation in Clouds, Internet and Networks (ICIN)*, pp. 136-140, 2022, doi: 10.1109/ICIN53892.2022.9758116.

[46] Y. Li et al., "Joint Placement of UPF and Edge Server for 6G Network," *IEEE Internet of Things Journal*, vol. 8, no. 22, pp. 16370-16378, 2021, doi: 10.1109/JIOT.2021.3095236.

[47] J. Song and M. Kountouris, "Wireless Distributed Edge Learning: How Many Edge Devices Do We Need?," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 7, pp. 2120-2134, 2021, doi: 10.1109/JSAC.2020.3041379.

[48] S. Yu et al., "When Deep Reinforcement Learning Meets Federated Learning: Intelligent Multi-timescale Resource Management for Multi-access Edge Computing in 5G Ultradense Network," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2238-2251, 2021, doi: 10.1109/JIOT.2020.3026589.

[49] A. Dalgkitsis, P. -V. Mekikis, A. Antonopoulos, G. Kormentzas and C. Verikoukis, "Dynamic Resource Aware VNF Placement with Deep Reinforcement Learning for 5G Networks," *IEEE Global Communications Conference*, 2020, doi: 10.1109/GLOBECOM42002.2020.9322512.

[50] A. Talpur and M. Gurusamy, "DRLD-SP: A Deep-Reinforcement-Learning-Based Dynamic Service Placement in Edge-Enabled Internet of Vehicles," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 6239-6251, 2022, doi: 10.1109/JIOT.2021.3110913.

[51] D. Li, M. Lan, and Y. Hu, "Energy-saving service management technology of internet of things using edge computing and deep learning", *Complex & Intelligent Systems*, vol. 8, no. 5, pp 3867–3879, 2022, doi: 10.1007/s40747-022-00666-0.

[52] Z. Yang et al., "Energy Efficient Federated Learning Over Wireless Communication Networks", *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1935-1949, 2021, doi: 10.1109/TWC.2020.3037554.

[53] M. Goudarzi, M. Palaniswami, and R. Buyya, "A Distributed Deep Reinforcement Learning Technique for Application Placement in Edge and Fog Computing Environments", *IEEE Transactions on Mobile Computing*, 2020, doi: 10.1109/TMC.2021.3123165.

[54] H. Zhou et al., "Energy Efficient Joint Computation Offloading and Service Caching for Mobile Edge Computing: A Deep Reinforcement Learning Approach", *IEEE Transactions on Green Communications and Networking*,, 2022, doi: 10.1109/TGCN.2022.3186403.

[55] Č. Stefanović, "Industry 4.0 from 5G perspective: Use-cases, requirements, challenges and approaches," *11th CMI International Conference: Prospects and Challenges Towards Developing a Digital Economy within the EU*, pp. 44-48, 2018, doi: 10.1109/PCTDDE.2018.8624728.

[56] H. Badri et al. "A Sample Average Approximation-Based Parallel Algorithm for Application Placement in Edge Computing Systems", *2018 IEEE International Conference on Cloud Engineering (IC2E)*, pp. 198-203, 2018, doi: 10.1109/IC2E.2018.00044.

[57] R. Yu, G. Xue and X. Zhang, "Application Provisioning in FOG Computing-enabled Internet-of-Things: A Network Perspective," *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pp. 783-791, 2018, doi: 10.1109/INFOCOM.2018.8486269.

[58] I. Alqerm and Jianli Pan. "DeepEdge: A New QoE-Based Resource Allocation Framework Using Deep Reinforcement Learning for Future Heterogeneous Edge-IoT Applications". *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 3942–3954, 2021. doi: 10.1109/TNSM.2021.3123959.

[59] P. Gazori, D. Rahbari, and Mohsen Nickray. "Saving time and cost on the scheduling of fog-based IoT applications using deep reinforcement learning approach". *Future Generation Computer Systems*, vol. 110 pp. 1098–1115, 2020. doi: 10.1016/j.future.2019.09.060.

[60] Y. Chen et al., "Data-Intensive Application Deployment at Edge: A Deep Reinforcement Learning Approach," *IEEE International Conference on Web Services (ICWS)*, pp. 355-359, 2019, doi: 10.1109/ICWS.2019.00064.