5G PPP Technology Board

AI and ML – Enablers for Beyond 5G Networks

Version 1.0

Date:	2021-05-11	Version:	1.0

DOI 10.5281/zenodo.4299895

URL http://doi.org/10.5281/zenodo.4299895

Executive summary

This white paper on AI and ML as enablers of beyond 5G (B5G) networks is based on contributions from 5G PPP projects that research, implement and validate 5G and B5G network systems.

The white paper introduces the main relevant mechanisms in Artificial Intelligence (AI) and Machine Learning (ML), currently investigated and exploited for 5G and B5G networks. A family of neural networks is presented which are, generally speaking, non-linear statistical data modelling and decision-making tools. They are typically used to model complex relationships between input and output parameters of a system or to find patterns in data. Feed-forward neural networks, deep neural networks, recurrent neural networks, and convolutional neural networks belong to this family. Reinforcement learning is concerned about how intelligent agents must take actions in order to maximize a collective reward, e.g., to improve a property of the system. Deep reinforcement learning combines deep neural networks and has the benefit that is can operate on non-structured data. Hybrid solutions are presented such as combined analytical and machine learning modelling as well as expert knowledge aided machine learning. Finally, other specific methods are presented, such as generative adversarial networks and unsupervised learning and clustering.

In the sequel the white paper elaborates on use case and optimisation problems that are being tackled with AI/ML, partitioned in three major areas namely, i) Network Planning, ii) Network Diagnostics/Insights, and iii) Network Optimisation and Control. In Network Planning, attention is given to AI/ML assisted approaches to guide planning solutions. As B5G networks become increasingly complex and multi-dimensional, parallel layers of connectivity are considered a trend towards disaggregated deployments in which a base station is distributed over a set of separate physical network elements which ends up in the growing number of services and network slices that need to be operated. This climbing complexity renders traditional approaches in network planning obsolete and calls for their replacement with automated methods that can use AI/ML to guide planning decisions. In this respect two solutions are discussed, first the *network element placement problem* is introduced which aims at improvements in the identification of optimum constellation of base stations each located to provide best network performance taking into account various parameters, e.g. coverage, user equipment (UE) density and mobility patterns (estimates), required hardware and cabling, and overall cost. The second problem considered in this regard is the *dimensioning considerations for C-RAN clusters*, in which employing ML-based algorithms to provide optimal allocation of baseband unit (BBU) functions (to the appropriate servers hosted by the central unit (CU)) to provide the expected gains is addressed.

In Network Diagnostics, attention is given to the tools that can autonomously inspect the network state and trigger alarms when necessary. The contributions are divided into network characteristics forecasts solutions, precise user localizations methods, and security incident identification and forecast. The application of AI/ML methods in high-resolution synthesising and efficient forecasting of mobile traffic; QoE inference and QoS improvement by forecasting techniques; service level agreement (SLA) prediction in multi-tenant environments; and complex event recognition and forecasting are among network characteristics forecasts methods discussed. On high-precision user localization, AI-assisted sensor fusion and line-of-sight (LoS)/non-line-of-sight (NLoS) discrimination, and 5G localization based on soft information and sequential autoencoding are introduced. And finally, on forecasting security incidents, after a short introduction on modern attacks in mobile networks, ML-based network traffic inspection and real-time detection of distributed denial-of-service (DDoS) attacks are briefly examined.

In regard to the Network Optimisation and Control, attention is given to the different network segments, including radio access, transport/fronthaul (FH)/backhaul (BH), virtualisation infrastructure, end-to-end

(E2E) network slicing, security, and application functions. Among application of AI/ML in radio access, the slicing in multi-tenant networks, radio resource provisioning and traffic steering, user association, demand-driven power allocation, joint MAC scheduling (across several gNBs), and propagation channel estimation and modelling are discussed. Moreover, these solutions are categorised (based on the application time-scale) into real-time, near-real-time, and non-real-time groups. On transport and FH/BH networks, AI/ML algorithms on triggering path computations, traffic management (using programmable switches), dynamic load balancing, efficient per-flow scheduling, and optimal FH/BH functional splitting are introduced. Moreover, federated learning across MEC and NFV orchestrators, resource allocation for service function chaining, and dynamic resource allocation in NFV infrastructure are among introduced AI/ML applications for virtualisation infrastructure. In the context of E2E slicing, several applications such as automated E2E service assurance, resource reservation (proactively in E2E slice) and resource allocation (jointly with slice-based demand prediction), slice isolation, and slice optimisation are presented. In regard to the network security, the application of AI/ML techniques in responding to the attack incidents are discussed for two cases, i.e. in moving target defence for network slice protection, and in self-protection against app-layer DDoS attacks. And finally, on the AI/ML applications in optimisation of application functions, the dash prefetching optimization and Q-learning applications in federated scenarios are presented.

The white paper continues with the discussions on the application of AI/ML in the 5G and B5G network architectures. In this context the AI/ML based solutions pertaining to autonomous slice management, control and orchestration, cross-layer optimisation framework, anomaly detection, and management analytics, as well as aspects in AI/ML-as-a-service in network management and orchestration, and enablement of ML for the verticals' domain are presented. This is followed by topics on management of ML models and functions, namely the ML model lifecycle management, e.g., training, monitoring, evaluation, configuration and interface management of ML models.

Furthermore, the white paper investigates the standardisation activities on the enablement of AI/ML in networks, including the definition of network data analytics function (NDAF) by 3GPP, the definition of an architecture that helps address challenges in network automation and optimization using AI and the categories of use cases where AI may benefit network operation and management by ETSI ENI, and finally the O-RAN definition of non-real-time and near-real-time RAN controllers to support ML-based management and intelligent RAN optimisation.

Additionally, the white paper identifies the challenges in view of privacy and trust in AI/ML-based networks and potential solutions by introducing privacy preserving mechanisms and the zero-trust management approach are introduced. The availability of reliable data-sets as a crucial prerequisite to efficiency of AI/ML algorithms is discussed and the white paper concludes with a brief overview of AI/ML-based KPI validation and system troubleshooting.

In summary the findings of this white paper conclude with the identification of several areas (research and development work) for further attention in order to enhance future network return-on-investment (ROI):

(a) building standardized interfaces to access relevant and actionable data,
(b) exploring ways of using AI to optimize customer experience,
(c) running early trials with new customer segments to identify AI opportunities,
(d) examining use of AI and automation for network operations, including planning and optimization,
(e) ensuring early adoption of new solutions for AI and automation to facilitate introduction of new use cases, and
(f) establish/launch an open repository for network data-sets that can be used for training and benchmarking algorithms by all.

Contents

List of figures

List of tables

1 Introduction

The fast adoption of 5G technology is promising a staggering number of new devices. For example the Cisco Annual Internet Report (2018-2023) [69] forecasts that “Machine-to-Machine (M2M) connections will grow 2.4-fold, from 6.1 billion in 2018 to 14.7 billion by 2023. There will be 1.8 M2M connections for each member of the global population by 2023”. The exponential growth in connected devices along with the introduction of 5G technology is expected to cause a challenge for the efficient and reliable network resource allocation. Moreover, the massive deployment of Internet of Things and connected devices to the Internet may cause a serious risk to the network security if they are not handled properly.

During the 5G era, network operators will have a chance to dynamically create and deploy different use cases or services such as massive Internet of Things (mIoT), massive Machine Type Communication (mMTC), Ultra-Reliable Low Latency Communication (URLLC), and enhanced Mobile Broadband (eMBB). This will be achieved via the concurrent support of several different logical networks (i.e., “network slices”) that will operate on top of the same physical infrastructure and will be fine-tuned to serve different requirements for different vertical sectors.

To tackle this level of flexibility and network complexity, service providers should come up with solutions to ensure the security, reliability and allocation of the necessary resources to their customers in a dynamic robust and trustworthy way. The use of Artificial Intelligence (AI) and Machine Learning (ML) as a key enabler for future networks has been recognized at European [9] and global level [206].

The identified challenges and corresponding opportunities given by AI/ML will affect different network aspects, layers, and functions and even create new requirements for the architecture of future mobile networks.

Despite the hype of the previous few years, the adoption of AI/ML methods in cellular networks is still at its early stages. A lot of work is still needed to identify the most suitable solutions for the dynamic network management and control via AI/ML mechanisms. Ongoing research activities need to take into consideration diverse aspects, such as the availability and usability of data sets needed for specification and testing of AI/ML solutions, regulatory aspects and practical implementation issues.

The aim of this white paper is twofold. Firstly it discusses at a high-level the potential applications of AI and ML mechanisms in 5G as well as Beyond 5G (B5G) and 6G networks. Secondly it presents in detail, how EU funded research projects, operating in the context of the 5G PPP Programme, have specified, developed, and tested specific AI/ML solutions.

This white paper is structured as follows. Section 2 provides an overview of AI/ML methods for network optimization, including the main principles of different approaches. Section 3 deals with optimization issues and use cases presenting specific solutions designed by 5G PPP projects, such as network planning (e.g., optimizing the placement of network objects), network diagnostics and forecasting of events including security incidents, optimization and control schemes for the different network domains as well as for end-to-end solutions including both computational and network resources. Finally, Section 4 discusses several architectural aspects starting from a brief overview of current activities in multiple standardization organizations and provides more detailed information about technical approaches, which are currently under investigation by 5G PPP projects.

2 AI/ML methods for network optimization – an overview

The recent paradigm shift that characterized mobile and fixed networks architectures allowed to evolve traditionally centralized and dedicated architectures into a common pool of resources, which can be dynamically orchestrated and tailored to service-specific requirements, e.g. in terms of communication latency and bandwidth. In this context, Artificial Intelligence (AI) is quickly becoming a key-feature in both network management and operational aspects of mobile networks. The wide availability of monitoring and operational data coming from heterogeneous networking domains allows gathering substantive insights on real-time networking processes. Decisions that previously took slow human interactions, based on traditional network characterization and optimization methods, can now be autonomously performed by Machine Learning (ML) algorithms with a holistic view of the network, enabling software components to directly contribute into decision-making activities related with the mobile network resource management. This not only improves the overall operational efficiency of the infrastructure, but also has significant impact into the reduction of management and energy related costs.

Despite the general applicability of ML-based solutions, their practical application often relies on the possibility to access real-time data to perform analytics and diagnostics. Most of the solutions available nowadays derive from the combination of few well-known frameworks. Therefore in the following, we will provide an overview of the existing and emerging ML frameworks as enablers for the adoption of Machine Learning solutions into the network management operations[1], as follows:

- Neural Networks
  - Feed-forward neural networks
  - Deep neural networks
  - Recurrent neural networks
  - Convolutional neural networks
- Reinforcement Learning
  - Basics/overview
  - Deep Reinforcement Learning
- Hybrid Solutions
  - Combined analytical and Machine Learning modelling
  - Expert knowledge aided Machine Learning
- Further Specific Methods
  - Generative adversarial networks
  - Kalman type filtering – and it relation to AI
  - Unsupervised learning and clustering

The latest ML developments assumes different neural network topologies distributed over multiple (hidden) layers. As topologies become more complex, deep-learning model benefits for training purposes from the adoption of Graphics Processing Units (GPUs) or programmable integrated circuits (FPGA) over common Central Processing Units (CPUs), mainly thanks to parallel computing platforms optimized for such intensive applications, e.g. the Nvidia CUDA package [119].

[1] Goal of this section is to present an overview on AI/ML methods in consideration by the networking community to be used for communications network design, planning, and optimization. It is not intended to provide yet another classification of the methods and/or to include/exclude any of the existing AI/ML approaches.

General machine learning tasks can be easily performed exploiting python programming language and its Scikit-learn library, which provides a wide selection of machine learning algorithms for classification, regression, clustering, dimensionality reduction, but lacks of methods to develop deep or reinforcement learning tasks. However, the majority of current applications require more advanced multi-layered ML-based models combining the best characteristics of each singular framework.

2.1 Neural networks

2.1.1 Feed-forward neural networks

Feed forward neural networks (FFNN), are a type of artificial neural networks and as so, the goal of FFNN is function approximation, i.e., to approximate function $y = f^*(x)$ in the best possible manner. To this end, FFNNs define a mapping θ and try to learn the best value of θ such that $f(x;\theta) \approx f^*(x)$. The label *feed forward* comes from the fact that information flow is unidirectional in a forward manner, starting from the input data x, going through the possible inner nodes, until reaching $y = f(x;\theta)$, such that the connections of its distinct nodes do not form a cycle.

Here, the term "networks" is used because the FFNNs are being compost of many different functions that are concatenated in chain. For example, $f(x) = f^2\left(f^1(x)\right)$ means that the entry values x, are first propagated through the first layer $f^1(x)$, and their outputs are processed by the second layer $f^2(x')$ to obtain an approximation of $f^*(x)$. The operations performed at each of the layers are usually defined by a weighted combination of each layer input units followed by a non-linear activation function. The typical activation functions are sigmoid functions, hyperbolic tangent, ReLU, etc.

Finally, FFNNs are also called neural as they are inspired from neuro-science, as each unit resembles a neuron as it can receive inputs from any other previous layer unit and it computes its own activation value that triggers and a stimulus into the network.

The learning of FFNNs is like any machine learning algorithm in the sense that a loss function given its parameters weights is computed, $J(\theta)$, and gradient decent is used to update the parameters and reduce the loss function. There are several types of loss functions, but the most utilized are the mean square error, the Huber Loss, Cross-entropy, the Kullback-Liebler Divergence, etc. [138].

The update of the parameters based on the obtained loss it is called the backward pass, and starts from the outer layer of the FFNN, which computes the loss, and it back propagates to the different layers of the neural network. Backpropagation updates the network parameters in an efficient manner by computing the loss function with respect to each weight using the chain rule, computing the gradient layer per layer, going backward from the last layer to avoid redundant calculations of intermediate terms.

In contrast with traditional regression methods, FFNNs reduce the need of expensive hand-crafted feature engineering as can automatically extract high-level features through layers of different depths. Furthermore, FFNN benefits from large amount of data, which is generally the case of mobile networks, as networks generates tremendous amounts of data and this data can be gathered from very distinct network scenarios, which helps to improve generalization.

On the other hand, as most machine learning techniques, FFNNs are perceived as black boxes due to their low interpretability. Furthermore, FFNNs are computationally demanding as tend to require very large networks (millions of parameters), which translates into high number of matrix operations that must be performed in both the forward and backward passes. Thus, these might not be a good approach for solution embedded onto mobile devices, where energy and computing power constraints have to be considered.

2.1.2 Deep neural networks

Artificial neural networks are computing systems inspired by animal brains and are based on a set of nodes, called artificial neurons that are interconnected by weighted unions, emulating brain synapses. In this sense, electrical signals are replaced by real numbers whose value is computed by a non-linear function as the sum of its inputs, and then this number is transmitted to the following neurons. To form a neural network, these nodes or neurons are grouped into sets called layers, so that the signal (real numbers) travels from one layer to another following a directed graph.

Thus, a neural network has 3 types of layers: input layer, where the data instances are received; hidden layers, through which the signal travels, undergoing modifications based on the weights of the connections and neurons; and output layer, which returns the result of the processing performed by the network. In this sense, a deep neural network (DNN) can be defined as a neural network that has more than one hidden layer [63].

DNNs are capable of modelling complex non-linear relationships by using data they receive as input, concretely for training. During this training process, connection and neuron weights are updated according to some pre-defined learning rules, usually doing backpropagation by calculating a cost function based on the error value regarding the expected model output. After this training, the DNN model is able to perform predictions in new unseen data using the intrinsic relationships learned during the training.

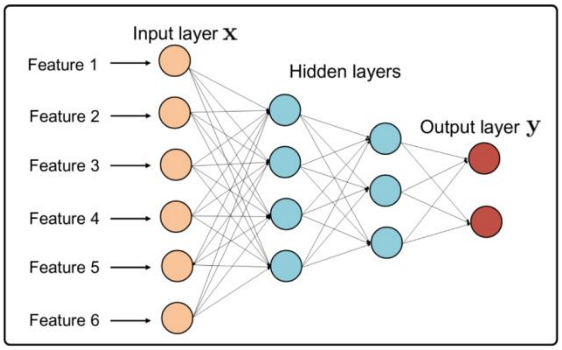


Figure 2-1 Scheme of a deep neural network with 2 hidden layers [297].

The most common architecture for DNNs is feed-forward, in which the information flows from the input layer to the output layer without internal loops. However, there are other types of architectures such as recurrent neural networks (RNNs), where the information propagates forward but also backwards to model time patterns, or deep belief networks (DBNs), which include bidirectional connections in some layers. Besides, depending on how the layers and neurons are structured, there are different types of networks such as multilayer perceptron (MLP), autoencoders, convolutional neural networks (CNNs), etc.

DNNs application for communication and networking optimization has increased in the last few years and will be even more extensive in the future [296]. Some areas where DNNs have settled state-of-the-art performance include power control and management in cellular and Ultra Dense networks [296], automatic computation offloading and edge caching [171], user location estimation, security incident forecasting, and intrusion detection. Furthermore, DNNs are also applied at the physical layer [253] for novel tasks such as spectrum analysis, radio virtualization and optimization, blockage prediction or beam alignment.

This wide application in network scenarios is motivated by the flexibility of DNN methods, which allows them to be adapted to practically any network scenario where data and metrics can be collected. In addition, the evaluation of new data using the generated models is fast, which makes it possible to deploy these solutions online and directly on the network elements.

The next subsections provide a deeper view in the main DNN architectures and methods, their properties, advantages, disadvantages, and their main applicability in network optimization problems.

2.1.3 Recurrent neural networks

Recurrent neural network (RNN) is basically a type of ANNs that is very effective in dealing with time-series data. It has a distinctive architecture that allows neuron connections from a neuron in one layer to neurons in previous layers, i.e., feedback from pervious layer(s). This seemingly simple change from typical ANNs, e.g., DNNs, enables the output of a neural network to depend on the current input as well as the historical input. This makes it very powerful in capturing dynamic temporal correlations such as those faced in mobility prediction and speech recognition. The combination of various activation functions and connection methods for the neurons define a set of RNN architectures such as echo state networks (ESNs), long short-term memories (LSTMs), and gated recurrent units (GRUs)[2].

In general, the most commonly used algorithm for training RNNs is the backpropagation through time algorithm [223]. However, RNNs require more time to train compared to traditional ANNs, since each value of the activation function depends on the series data recorded in RNNs (due to the feedback between layers). To reduce the training complexity of RNNs, ESN is developed in a way that it needs only to train the output weight matrix. Due to the ESNs' appealing properties such as the capability of capturing temporal correlation and training simplicity, they have been widely applied for various tasks such as supervised learning, RL, classification, and regression.

We are convinced that the most promising applications of ESNs for future networks will include content caching and DRL-based network optimization. In principle, ESNs can be used to predict the content request preference of network users based on their historical data. Since the users' behaviour data such as content request is temporally-correlated, an ESN-based model can quickly learn user request distributions and thereby predicting the future content demand. For this application, the input of an ESN model might include user's contextual information such as gender, occupation, age, and device type, whereas the output would be the prediction of a user's content request. ESNs can also be leveraged to empower the reinforcement learning (RL) framework with the capability of capturing temporal correlations inherent in sequential data. It is worth mentioning that RL is a very powerful technique that is commonly used in network optimization applications such as proactive resource allocation, e.g., see [67].

Among all types of RNN models, LSTM stands out as an ideal candidate for a wide range of applications concentrating on network traffic forecasting, prediction, and recommendation [299]. At the time of this writing, LSTM is the most widely adopted model for traffic forecasting, making it the de-facto tool for this kind of applications. Moreover, forecasted information pertinent to traffic and cloud price can be further leveraged for proactive network optimization, e.g., in advance allocation of resources such as computing, memory, bandwidth, etc. In detail, tradition RNN models generally experience a vanishing gradient problem which impedes learning of long data sequences. This is because when the gradient becomes smaller, the RNN parameter updates become intangible, which hinders the learning process. To overcome this issue, LSTM models have a gated structure composed of three gates, namely, forget gate,

[2] GRU use less training parameters and therefore use less memory, execute faster and train faster than LSTM's whereas LSTM is more accurate on dataset using longer sequence. In short, if sequence is large or accuracy is very critical, LSTM is the solution whereas for less memory consumption and faster operation GRU is better to use.

input gate, and output gate. These three gates solve the vanishing gradient problem of RNNs by collectively controlling which information in the cell state to forget, given new information entered the network, and which information to be mapped to the network output.

2.1.4 Convolutional neural networks

Generic Convolutional Neural Networks (CNNs) [164] are a specialized kind of deep learning structure that can infer local patterns in the feature space of a matrix input.

In particular, two-dimensional CNNs (2D-CNNs) have been extensively utilized in image processing, where they can complete complex tasks on pixel matrices such as face recognition or image quality assessment [274]. Each neuron or filter process completely the input but piece by piece instead of in a one-shot manner. The size of the piece is given by the kernel size.

3D-CNNs extend 2D-CNNs to the case where the features to be learned are spatiotemporal in nature, which adds the time dimension to the problem and transforms the input into a 3D-tensor. Since mobile network traffic exhibits correlated patterns in both space and time, due to the intrinsic human nature of network utilization, the usage of 3D-CNN can be used to infer patterns from a matrix of network demands time series, which have a spatial components (base stations are geographically located on a 2D plane, and a temporal one. An example of such solution is described in section 3.3.4.

Having discussed the concept behind CNNs and RNNs, we now shed light on an extension model that is yielded by combining RNN and CNN models together, namely, spatio-temporal network model [304]. It is now clear that time-series data such as the traffic of a network base station experiences temporal correlation. In addition to this temporal correlation, spatial correlation might be encountered between network nodes, e.g., base stations that are deployed in similar environments. For example, base station deployed in train stations in a city might have both temporally and spatially correlated traffics. To jointly capture such spatio-temporal correlations for applications such as prediction and traffic forecasting, convolutional-LSTM models based on CNNs (spatial correlation) and LSTM (temporal correlation) can be leveraged.

2.2 Reinforcement learning

Reinforcement learning is a class of solution methods, where solutions are learned through trial-and-error, i.e., an agent learns to perform actions in an environment by interacting with it and receiving feedback regarding the performed actions. In contrast with many forms of machine learning, the learner is not told which actions to take, but instead, must discover which actions yield the most reward by trying them out (unsupervised learning). The goal of the agent is to maximize its cumulative reward, also referred to as expected return. Different reinforcement learning methods yield distinct behaviours for the agent to achieve their goal. This type of solutions has drawn the attention to mobile network researches due to its proven efficacy to address complex multi-domain problems yielding close to optimal results.

Most reinforcement learning problems can be formulated as Markov Decision Process (MDP), where at each interaction with the environment an agent observes a state $s^{(t)} \in S$ (which might be partially observable o), where S is the state space and selects an action $a^{(t)} \in A$, where A is the set of all possible actions. Action $a^{(t)}$ in state $s^{(t)}$ receives a certain reward $r^{(t)} \in R$, where R: S × A → R denotes the reward function, and the environment transition to a new state $s^{(t+1)} \in$ S with probability $p(s' \mid s, a) \in$ P, where P|: S×A×S → [0,1] is a probability kernel. At each interaction, the agent maps the observed state $s^{(t)}$ to a probability distribution over the actions set A, this mapping is referred to as policy, and it is denoted by π. The probability of selecting action $a^{(t)}$ in state $s^{(t)}$ is given by $\pi\left(a^{(t)}|s^{(t)}\right)$. The goal of the agent is to determine the optimal policy $\pi^* \in \Pi$ that maximizes the obtained reward overall admissible policies, which is called expected return. The *state-value* function $V_\pi(s)$ for policy π is defined as the expected return that the agent would accumulate after being at the state s and following policy π afterwards. Similarly, the *action-value* function $Q_\pi(s, a)$ for policy π is defined as the expected return the agent would accumulate starting at state s, taking action a, and following policy π afterwards. The fact that most mobile network problems (such as resource allocation, slice management, etc.) can be formulated as MDPs makes this type of solutions a powerful tool to explore.

We proceed to provide an overview of the landscape of methods used to maximize the expected return. The first classification that one encounters when learning about RL methods are the *Model-Based* and *Model-Free* taxonomy.

- *Model-Based algorithms*, aim to learn a model of the environment i.e., learn the transition probabilities $p\left(s^{(t+1)}, r^t \middle| s^{(t)}, a^{(t)}\right)$ which results in an estimated MDP model. Once the agent has adequately modeled the environment, it can use a planning algorithm given its learned model of the environment to find a policy that maximizes the expected return. Among the most popular algorithms in this category we highlight the well-known work of AlphaGO [269].
- *Model-Free algorithms* do not intend to learn a model of the environment but to learn either the state-value or action-value functions, from which a policy may be derived, or the policy itself, using a policy search method that directly searches over the policy space to maximize the expected return.

The model-based algorithms have found limited application to the mobile communication problems as network deployments tend to be too complex to model, and thus, finding these transitions probabilities and planning ahead seems, as of now, unfeasible. However, the model-free class of solutions have found widespread application and most of the solutions that leverage RL in communication networks belong to this category [297]. More details of the methods for model-free RL are provided. This category can further be broken down into:

- Value-based Methods. In this family of methods, an estimate of the expected return for each of the action-state value function $Q_\pi(s, a)$ is tracked and a policy is selected based on these estimates, e.g., an ε-greedy policy might be selected where with probability ε $(0 \le \varepsilon \le 1)$ an action is chosen at random and with probability $1 - \varepsilon$ the action with the highest $Q_\pi(s, a)$ is chosen. Typically, an objective function is to be optimized by means of Bellman equations. Among the value-base methods, the most well-known are *Q-learning* and *SARSA* [273].

  These types of algorithm have been proven to work very well on mobile network domains that are limited to discrete action selection. For example, binary cell on-off decision algorithms have been proposed using Q-learning. The aim is to achieve interference mitigation of same-frequency cells by selecting which co-existing cells are turned off and on such that interference is minimized. Further applications can be found on mobile edge computing, where the binary decision of offloading a task to the edge or executing on the device has to be made.

- Policy optimization. Methods in this family present a parameterized policy $\pi_\theta(a|s)$ and they optimize the parameters θ by directly applying gradient ascent on a performance objective function (π_θ) . The most prominent methods in this category are vanilla *policy gradient* and *PPO* [273].
- Actor-critic methods [273]. The actor critic method is a combination of the previous methods as it learns both, the action-value functions, and a policy. In this method there are two sets of parameters, θ_1 and θ_2, parameterizing the actor and the critic respectively. The actor represents a

parameterization of the policy and the critic parameters are used to estimate the action-value pairs. To train the parameters, the actor selects the actions given the state and these actions are then fed to the critic, which evaluates these actions and update its action-values estimates accordingly. Then, this evaluation is passed back to the actor, who updates the policy parameters in the direction suggested by critic.

These methods are very powerful tools, suitable for complex environment settings like resource allocation and orchestration, VNF deployments, etc. The Model-free approaches approach has a complete exploration of the environment, however, for this reason, when applied to complex systems like 5G networks, the learning phase can be very inefficient, requiring a considerable amount of time before reaching optimality. Novel ML techniques such as Deep Q-learning (DRL), which approximate the Q-values with Deep Neural Networks (DNN), can overcome this issue and enable a complete exploration minimizing the approximation loss of DNN.

2.2.1 Deep reinforcement learning

Traditional Reinforcement Learning methods struggle to address real-world problems due to their high complexity. In these problems, high-dimensional state spaces need to be managed to obtain a model that can generalize past experiences to new states. Deep Reinforcement Learning (DRL) aims to solve this problem by employing NNs as function approximators to reduce the complexity of classical RL methods.

In [197] authors introduce deep Q-learning network (DQN), where a DNN is used as a function approximator for action selection on a discrete action space, based on Q-learning. Given a state, Q-learning updates the action-value estimate with the immediate reward plus a weighted version of the highest Q-estimate for the next state. Using a combination of 3 convolutional layers (for computer vision) and two fully connected layers (Q-learning part), they obtain human-level results for a wide range of Atari games.

To overcome the limitation of discrete action selection, in [281] the idea of DQN is extended to continuous action spaces using the deterministic policy gradient (DPG) theorem, in particular the deep-DPG (DDPG) method. DDPG extends the use of DNN to the actor-critic method leveraging off-policy learning, where a deterministic policy is learned using a combination of *replay buffer* and *target networks* to ensure stability and a zero-mean Gaussian noise is added to the actions for action space exploration.

The fact that mobile networking problems can be formulated as MDPs, where reinforcement learning can be used to obtain optimal solutions, has drawn the attention of the network research community in both, academia and industry, to investigate and apply DRL solutions. Most notable applications can be found in the use of DRL for base station on- off switching strategies [250], optimal routing [248], and adaptive VNF MANO [136]. Not surprisingly, traditional RL had not been applied to communication networks until now, as most of these problems involve high-dimensional inputs, which limits the applicability of traditional reinforcement learning algorithms. The inclusion of DNNs techniques improve the ability of traditional RL algorithms to handle high dimensionality, in scenarios previously considered intractable. DRL is thus a promising field of study to address network management and control problems under complex, changeable, and heterogeneous mobile environments. DQN have found wide application in communication networks, from mobile edge computing [293], to network slicing [60] to name a few.

The main drawback of this type of solutions is the high number of environment interactions. In order to explore all possible actions and its possible outcomes, a lot of trajectories (state, action, reward, and transition states tuples) have to be gathered from the environment, which is a big price to pay when models are to be deployed in a production environment. To overcome this type of limitation, realistic environments simulating close to reality networks can be employed to train and test different algorithms' performance.

2.3 Hybrid solutions

2.3.1 Combined analytical and ML models

5G and beyond mobile networks, have the ambition to achieve zero-touch network and service management, i.e., the full automation of the system. One of the critical ingredients to cost-effectively realizing such a scenario is to devise trusty and smart mechanisms to efficiently handle the involved management operations such as instantiation and auto-scaling of network services or flow scheduling (e.g., traffic steering, packet replication, and resource reservation). Ultimately, those operations require the translation of end-to-end Quality of Service specifications into specific actions such as reserving or releasing a certain amount of resources, choosing the concrete physical machines to embed the virtualized services, or allocating an explicit path for a stream.

Traditionally, analytical performance modelling has been used extensively to assist network planning, management, control, and maintenance. Briefly, analytical performance modelling consists of deriving a set of equations relating the performance metrics of interest (e.g., end-to-end delay, jitter, packet loss probability, and reliability) with some features of the network (e.g., topology, nodes hardware features, physical distances between nodes and available resources) and traffic demands (e.g., packet arrival rate, degree of autocorrelation, and coefficients of variation) either theoretically or empirically. If required, the derived models might be reversed, for instance, by integrating them in the formulation of an optimization problem targeted to make a given decision. Nonetheless, this approach has the following three drawbacks:

- Generally, it requires high domain knowledge to address the problem.
- The derivation of an accurate model for some scenarios might be complicated or even impossible. Please consider, for example, feedback and lossy networks with stochastic service processes and self-similar and long-range dependent arrival processes.
- Finding a solution for the resulting optimization problem might be computationally complex or even intractable. By way of illustration, most of the decision versions of the resource allocation problems are NP-Complete.

Furthermore, the increasing complexity of the upcoming mobile networks, which combine a myriad of different technologies, each one offering significant flexibility within a large space of possible configurations, will intensify the problems referred to above, making the analytical model-based approaches impractical.

As mentioned in the previous subsections, ML has been enshrined as a useful approach to tackle future 5G and beyond networks' complexity. However, ML raises new concerns over the system stability, performance guarantees assurance, the amount of data required to produce a ready-to-use solution, and data availability.

Using ML and analytical model approaches in synergy might alleviate their shortcomings and fit the decision-making problems' necessities in many scenarios, as supported by previous works [32], [134]. The authors in [32] propose to harness the existing accurate analytical models for assisting the training of ML models intended to make optimal decisions. More precisely, analytical models serve to simulate the behaviour of the network with great agility. In this way, measurement campaigns might be avoided, speeding up the training process [32]. Simultaneously, the ML model enables to configure networks in real-time with near-optimal performance. As stated in [32], some works have reported and highlighted a drastic reduction in the amount of data required to train the ML-based models by employing that hybrid approach even when the available analytical model is not accurate.

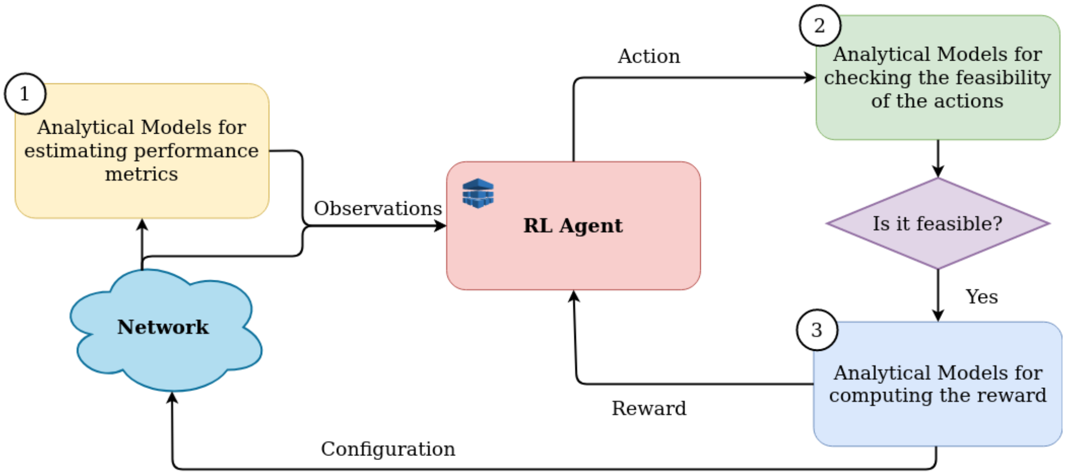

1
Analytical Models for estimating performance metrics
Network
Observations
RL Agent
Action
2
Analytical Models for checking the feasibility of the actions
Is it feasible?
Yes
3
Analytical Models for computing the reward
Reward
Configuration

Figure 2-2 Analytical models to assist Reinforcement Learning

The authors in [134] identify further advantages of the combined use of ML and analytical models. They consider the reinforcement learning framework to illustrate them, as shown in Figure 2-2, though the same advantages can be extrapolated to other ML techniques. First, some performance metrics are difficult or unfeasible to measure directly from the network, like the worst-case delay. Then, analytical models can be used to estimate those unavailable data. This approach might be useful, for instance, to provide some inputs to the RL Agent as depicted in Figure 2-2, or to compute relevant features for supervised ML. Second, as ML models are not 100% accurate, they might issue unfeasible decisions, i.e., decisions that do not fulfil the optimization problem's constraint. If the network is configured according to such a decision, undesired behaviour, such as SLA violation, might occur. We can leverage analytical models for such a feasibility check in this vein, thus making the solution fully reliable. Then, it is assured that only valid configurations of the network will be applied. Last, analytical models might be used to assess the degree of optimality of the ML model's action to compute an RL agent's reward.

2.3.2 Expert knowledge aided ML

Every aspect of past and present wireless communication networks is regulated by mathematical models, which are either derived from theoretical considerations, or from field measurement campaigns. Mathematical models are used for initial network planning and deployment, for network resource management, as well as for network maintenance and control. However, any model is always characterized by an inherent trade-off between accuracy and tractability. Very complex scenarios like those of future wireless networks are unlikely to admit a mathematical description that is at the same time accurate and tractable. In other words, we are rapidly reaching the point at which the quality and heterogeneity of the services we demand of communication systems will exceed the capabilities and applicability of present modelling and design approaches.

A recent trend in deep learning complements purely data-driven approaches with prior information based on expert knowledge. Traditional deep learning methods acquire a large amount of empirical data and employ it to perform optimization. However, the application of deep learning to communication network design and optimization offers more possibilities than such a purely data-driven approach. Although mathematical models for communication network optimization may be simplified and inaccurate, they are very often available, which is not the case for other fields such as image classification and speech recognition. This a priori expert knowledge, acquired over decades of intense research, should not be dismissed and ignored.

In several prior works, machine learning has been employed under simplified settings while adoption aspects also lack due attention [63]. In this direction, we try to address some of these shortcomings within the B5G network management landscape. System modelling problems, joint UE-AP association and resource allocation strategies, as well the user mobility for proactive hand-over and continuous adjustment are only some of the problems that are addressed. Our goal is to explore and apply AI techniques for automated decision making, their potential when used in synergies or in liaison to attack the problem at hand in the most befitting manner. This includes constraint solving optimizations, machine learning (ML), but also conventional and state of the art in AI to examine the unique capabilities these approaches deliver. Next, we present the indicative example of system modelling, joint user association and resource allocation in sub-THz and THz wireless systems as well as the need of providing novel AI-based mobility management solutions.

System modelling: In [32] and [33], the authors report two main cases in which expert knowledge and data-driven based design may nicely complement each other: (i) in the presence of a model deficiency, data-driven method may be used to refine approximate models; and (ii) in the presence of an algorithmic deficiency, AI-based methods may be trained by using data from sufficiently accurate models. It is shown in [32] and [33] that the two case studies can be merged by leveraging the tools of transfer learning and deep unfolding.

User association and resource allocation in sub-THz and THz wireless systems: The ultra-wideband extremely directional nature of the sub-THz and THz links in combination with the non-uniform user equipment (UE) spatial distribution may lead to inefficient user association, when the classical minimum-distance criterion is employed. Networks operating in such frequencies can be considered noise- and blockage-limited, due to the fact that high path and penetration losses attenuate the interference [18], [17], [90]. Hence, user association metrics designed for interference limited homogenous systems are not well suited to sub-THz and THz networks [20]. As a result, user association should be designed to meet the dominant requirements of throughput and guarantee low blockage probability. Additionally, user orientation has an important impact on the performance of THz links [16], [89], [15], [19]. As a consequence, users may not be associated with the geographically closest access point (AP), since a better directional link may exist for a farther away one. Furthermore, the network needs to predict users and blockers movements in order to proactively hand-over the UE to another AP. From the technical point of view, to deal with the aforementioned requirements two type of mechanisms need to be devices, namely joint user association and resource allocation and proactive hand-over.

Scanning the open technical literature, several joint user association and resource allocation policies can be identified [306], [161], [222]. In particular, in [306], the authors reported an online deep reinforcement learning (DRL) based algorithm for heterogeneous networks, where multiple parallel deep neural networks (DNNs) generate user association solutions and shared memory is used to tore the best association scheme. Similarly, in [161], the authors presented a federate learning approach to jointly minimize the latency and the effect of loss in the model accuracy due to channel uncertainties. Finally, in [222], a deep deterministic policy gradient based algorithm was employed to solve the joint user association and resource management problem in mobile edge computing. All the aforementioned approached come as a solution of optimization problems and, for the sake of simplicity, they neglect the impact of dynamic blockage due to moving obstacles as well as the influence of beam orientation errors, due to end-users movement.

To counterbalance this, we employ artificial intelligent (AI)-based approaches that return the outage probabilities between each one the UEs and the APs, due to UE or blocker movement. Based on these probabilities as well as the UE data-rate demands, we formulate a joint user association and resource allocation problem that maximizes the networks throughput, while low outage probability. In more detail, we model several representative and demanding user association scenarios and generate datasets - upon which optimizations are performed to find the best UE-AP association-allocation solutions. These solutions respect the hard and soft constraints in dense and sparse network topologies as foreseen by domain experts for B5G networks. Constraints consider UE and AP locations, the required and available bandwidths to be allocated in respect of requested data rates, where the aggregate data-rate or throughput needs to be maximized. To address some gaps in the current literature, we pay due diligence to blockade schemes that limit UEs in establishing Line of Sight (LoS) links with APs due to their location (polar coordinates). These schemes include: i) full blockades where the UE's association space is reduced to unblocked APs and their resource capacities, ii) static blockades, where the UE's LoS is hindered by fixed physical blockades and iii) partial blockades, which account for cases where an LoS may be partially possible and depending on signal properties, UEs may still be associated to APs if required QoS thresholds may be met. The inability to find effective LoS-aware user-association and allocations serves as a pathway to non LoS solutions, which are mentioned in section 3.2.2.4.

In recent works, such as [14] and [137], some hybrid approaches where model-based and data-driven concepts are used together are seen. Considering the potential of such approaches, we devise a roadmap, where at the first stage, we devise experiments to frame and solve above mentioned resource allocation problems using metaheuristic constraint solving methods and move on to bridge this heuristic approach to predictive and prescriptive analytics. Early results from our analysis reveal the combinatorial hardness of these problems, where the large state spaces range up to 10^{29025} or higher, and the challenge to discover effective assignments is limited not just by the used hardware but also the features of the used technology. For instance, in the heuristics domain, solving multi-constraint problems having less than a 2% required-to-available resource ratios, the hard and soft constraints need to be implemented using delta-score calculation (e.g. by incorporating a rule engine as we do) to speed up calculation speed by only calculating changed variables and thus exploring more of the state space in the same time period. The discovered optimization solutions (assignments) are used to complete the dataset i.e. the assignments for a given network topology lead to the identification of labels (attributes to predict), which allow to apply supervised machine learning approaches. It is to be noted that identification of labels is one of the most important steps in predictive modelling. In the mentioned problems, training data can be prepared in a couple of ways, which determines the choice of predictor to either: i) an ensemble of single-label predictor models, which is the widespread approach, or ii) a multi-label predictor model, which considers label chaining (hierarchy of correlation among individual label predictions).

Mobility management in sub-THz and THz wireless systems: After associating UEs to APs, uninterrupted connectivity needs to be guaranteed. However, as also noted earlier, the network is continuously undergoing change, hence the network management should be adaptive as well. This is where the conventional heuristic based exploration of state space needs to be extended to simulate UE mobility in online manner. The area of online optimizations is not exhaustively researched in prior art and there is ample room for advancements. The UE mobility partially modifies the network topology in real-time as the polar coordinates of UEs change, which also changes blockades faced with respect to APs. Thus, the online aspects of optimizations are also being investigated by considering discrete event simulation techniques that can be tweaked to simulate different scenarios. To deliver un-interrupted service to the UE despite various rates of mobility, the joint association-allocation solution scheme must recover from its infeasible state to the new feasible state for the updated topology. As presented earlier, one direction of work looks at the mobility prediction and pro-active hand-over, so controls are in place to prevent spontaneous overload and service interruption.

Another direction being investigated is the liaison of real-time optimization in which the heuristic approach continues to stabilize association-allocation infeasibility in real time, while in parallel, the ML model already trained for the offline case can be continuously updated to learn the newly discovered feasibility patterns in real time. This scheme also provides for better comparison of the two approaches. The new incoming (and previously unseen) data introduces variation in the already seen data. This variation reflects a new snapshot of the network, for which the predictive models may not be able to predict accurately if the distributions are not close enough to the mean values seen in the training data. In real time deployments, autonomous systems often run into such risks, where locally sensed data that may contain potential outliers, may affect the global pattern learned by the model - if not proportionally treated. Hybrid approach (liaison of heuristic and predictive analytics) provides a safer fall-back if model update requires time and inspection. It also provides a qualitative and quantitative schematic to test the limits of the used technology which are important concerns for operationalisation. This includes functional properties (i.e. correctness, which can be measured as a solver-specific heuristically-discovered solution’s score or a predictive model’s accuracy or another performance metric) and non-functional properties (i.e. response time of the heuristic to restore the system from infeasible to feasible state, or the update or retraining time of the predictive model and its inference/scoring time).

There is an increased focus on the application of various artificial neural network architectures (ANNs) in the network management literature. Within the field of Deep Learning, various ANNs have been shown to deliver promising results [63]. One such approach is Deep Unfolding, which is reported in recent art, where Recurrent Neural Network or Deep Transfer Learning models are trained on solutions discovered by greedy or approximate algorithms in a time-demanding fashion [88], [32]. In the joint association-allocation problems presented here, we foresee the application of Deep Transfer Learning among others, where hidden layer(s) can be trained on combinatorically-explored solution data discovered by the heuristic algorithms and offloading the discovery of irregularities and non-linear relationships in data to the model training phase. The objective is to reduce the combinatorial problem to a predictive problem, which are not just efficient in instant predictions for association and allocation, but also allows for a framework level approach to continuously update and improve the model to the point that the network management can be performed autonomously by the machine. This work opens further avenues, which Ariadne would aim to address in its roadmap. One such promising direction is the use of Reinforcement Learning, which does bring several presented concepts into a unified scheme, with the differentiator being the use of stochastic models as well borrowing from the paradigm of agent-oriented programming.

Another approach that we examine is the development of proactive hand-over mechanisms need to be utilized. An important amount of research effort was put in this are in the last couple of years. In more detail, in [283], the authors presented a centralized reinforcement learning (RL) method to maximize long-term utilities in millimetre wave (mmWave) networks. Similarly, in [131], a Bayesian regression based policy was introduced for low-frequency high-speed railway systems. Meanwhile, in [92], a machine learning (ML) based proactive, handover algorithm that employed multiple metrics to predict the future state of the network and optimize the load in order to ensure preservation of the quality of service (QoS) and experience (QoE) was proposed, whereas, in [260], linear regression, long-short term memory, and recursive neural networks methodologies were examined as possible approaches for network load prediction and a proactive hand-over policy was presented. Finally, in [301], the authors presented a deep reinforcement learning (DRL)-based hand-over management algorithm to address the large-scale load balancing problem for ultra-dense networks. Despite the importance of UE orientation and blockage probability in sub-THz and THz networks, to the best of our knowledge, there exist no contribution that present a generalized hand-over mechanism. Motivated by this, we are going to analyze the mobility management problem and present ML-based solutions.

2.4 Promising ML techniques for networking problems

2.4.1 Generative adversarial networks

Generative adversarial networks (GANs) [312] are deep learning models whose main objective is to generate realistic new data samples with the same properties that the training data. These models use two neural networks in a combined way, one called generator and another called discriminator.

In a summarized way, the generator network is in charge of creating new data, also known as "fake samples", whose features have a distribution similar to the training data, while the discriminator network is in charge of differentiating the data generated by the generator from the real samples. When training begins, the generator produces obviously fake data, and the discriminator quickly learns to tell that it's fake, but the data generated then is perfectioned to fool the discriminator. In this sense, both networks "indirectly" train each other, so when the generator improves the quality of the output samples, the discriminator improves its capacity to differentiate real and fake samples, and vice versa.

The main advantage of these models lies in their ability to generate data with a quality, at a realistic level, much higher than other generative models such as Variational Autoencoders. In addition, the discriminator network can be used directly as a classifier of false and real data. Besides, these networks can handle high dimensional spaces in a much more efficient way than other methods such as Boltzmann machines. However, as drawbacks, training this type of model is often complex because of instability in the combined network training process, and it requires a high number of computational resources. In addition, false patterns can be generated when working when discrete data, such as text.

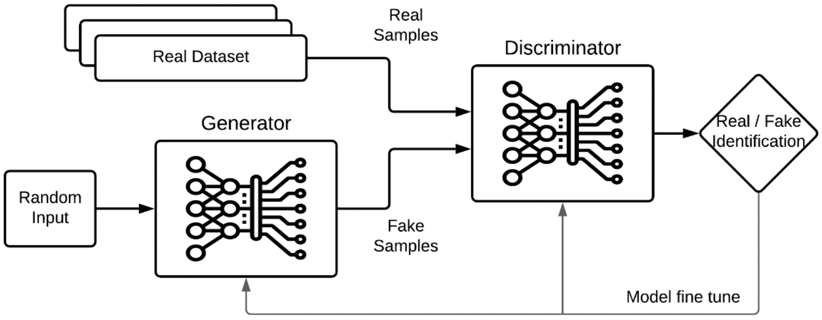


Figure 2-3 GAN topology scheme

These models have been applied for network optimization from several perspectives, being its applicability highly relevant in 5G self-organizing networks [128]. These modern networks require exhaustive labelled data to train the models in charge of automatic network management. However, low data comes already labelled from direct network monitoring, and the labelling process can be highly expensive and slow. In this context, realistic synthetic data generated using GANs is very relevant, as it can help to solve the previous issue, increasing the amount of data available at a low cost.

In this sense, GANs allow inferring fine-grained mobile traffic maps from coarse-grained, which are collected by traffic monitoring probes. As the traffic monitoring probes cannot be placed in every network point, the data that are gathered for traffic maps are not enough for the purpose of training a complex neural network. The presence of fine-grained mobile traffic maps allows the prediction of anomalous events in mobile traffic, such as network congestion, burst prediction [284] or traffic pattern recognition

[172]. This allows the fast response of MNOs in order to enable early countermeasures, such as the reallocation of UEs in different cells, improving resource usage and planning.

Besides, QoS management has also been improved by GANs [294] enhancing the prediction of metrics such as delay, packet loss, jitter, etc. GANs have been applied not only in traffic-related scenarios but in other key aspects in 5G networks, such as call frequency and duration forecasting [128]. In the physical layer, Massive MIMO antennas management has been also improved using these models for channel state information generation. Similarly, cell coverage planning and performance optimization has been also enhanced using GAN models [186].

2.4.2 AI-enabled network tomography

Network tomography (NT) has been proposed as a methodology for the efficient inference of network performance characteristics based on measurements realized at a subset of accessible network elements. It can be broadly classified into the following three categories [323]: 1) link-level NT; that regards the estimation of per link QoS parameters (e.g., loss rates, delays, jitter) based on end-to-end path measurements, 2) path-level NT; that concerns the estimation of the origin-destination traffic intensity matrix based on link-level measurements, and 3) topology inference; for reconstructing an unknown network topology. Compared to conventional monitoring techniques involving direct measurement of all objects of interest, NT alleviates the need for special-purpose cooperation of all devices and reduces the measurement traffic overhead.

NT belongs to the class of statistical inverse problems and can be formulated as a system of linear equations, $\mathbf{Y} = \mathbf{AX}$, where $\mathbf{Y}$ is the vector of observed measurements, $\mathbf{A}$ is the routing or measurement matrix representing the network topology, and $\mathbf{X}$ is the vector of unknown performance parameters. The goal is to estimate the unobserved vector $\mathbf{X}$ given the aforementioned linear model and the known vector of measurements $\mathbf{Y}$. From the perspective of linear algebra, $\mathbf{X}$ is uniquely identifiable if and only if the number of equations equals the number of the vector's components (unknown variables). However, $\mathbf{A}$ is usually an ill-posed matrix (i.e., the linear system of equations is under-determined) and, hence, non-invertible. Lately, the potential of enhancing NT with AI/ML has been suggested for relaxing the assumptions and the statistical modelling techniques typically employed for overcoming the ill-posed feature of NT and exploiting the bulk of measurement volumes available in the current and future network infrastructures.

In greater detail, an AI-based approach of solving the inverse problem at hand could employ deep neural networks (DNNs) for minimizing the L_2 error norm $\left\|\mathbf{X} - g_{\varphi}(\mathbf{Y})\right\|^2$, where $g_{\varphi}(\cdot)$ is a suitable function that assumes the role of $\mathbf{A}^{-1}$ (which cannot be precisely modelled mathematically) and is designed to correspond to a DNN with parameters φ that are learned from large data sets containing pairs of examples ($\mathbf{Y}$, $\mathbf{X}$). Such a learning procedure leads to a direct mapping of $\mathbf{Y}$ to $\mathbf{X}$ and shifts the computational burden to the learning phase, since providing an estimation of $\mathbf{X}$ for a given $\mathbf{Y}$ is represented by a feed-forward network $g_{\varphi}(\cdot)$, thus, it is computationally efficient. In other words, the choice of a specific neural network architecture $g_{\varphi}(\cdot)$ indicates the set of functional relationships that must be learned by training the DNNs in a supervised fashion over data sets containing a great number of pairs ($\mathbf{Y}$, $\mathbf{X}$). Given the formulation of the inverse problem as a regression problem, the MSE cost function can be employed and the parameters φ of the model can be iteratively updated by an optimization algorithm (e.g., stochastic gradient descent). After the optimal parameters are obtained, $\mathbf{X}$ can be estimated from the observed $\mathbf{Y}$ by using the trained network: $\widehat{\mathbf{X}} = g_{\varphi}(\mathbf{Y})$.

An example of employing a deep learning-based NT method for inferring a traffic matrix from the available link counts and routing information is presented in [208], [209]. The authors use a deep belief network (DBN) architecture in order to learn the properties of the ill-posed inverse system. Assuming a network with N nodes and L links, the proposed deep architecture is trained using prior measurements of link counts (L-by-$\tilde{T}$ matrix $\tilde{\mathbf{Y}}$, with $\tilde{T}$ timeslots) as input and the corresponding TM (N^2-by-$\tilde{T}$ matrix $\tilde{\mathbf{X}}$, with $\tilde{T}$ timeslots - row j corresponds to the time series of origin-destination pair j) as output to extract the mapping from $\tilde{\mathbf{Y}}$ to $\tilde{\mathbf{X}}$. It should be noted that, in this formulation, routing matrix **A** has dimensions L-by-N^2. Then, the traffic matrix **X** can be estimated by the trained model using the corresponding **Y** as input data. Simulation results show an estimation error improvement ratio of at least 23.4% compared to other conventional NT approaches. Another method based on a back-propagation neural network (BPNN) that accepts the vector of link loads as input and estimates the origin-destination flows in an output vector is described in [141]. More recent works have attempted to extend the input of the employed neural networks (BPNN [315], or convolutional neural network [93]) by including routing information, either implicitly in the form of the Moore-Penrose pseudoinverse of the routing matrix [315], or explicitly using graph embedding [93].

2.4.3 Kalman type filtering

The Kalman filter (KF) presented in 1960 [149] is a recursive solution that efficiently and robustly estimates the state and error covariance of a discrete time-controlled process [288], [289]. The KF is robust, given its ability to provide reasonable estimations of past, present, and future states with noisy, indirect or altogether missing measurement data. Generally, the KF considers weightings of measurement uncertainties to determine an optimal state output and its error covariance. Additionally, the KF lends itself to data fusion between two or more measurement sources, adjusting their influence on the result through weighting their individual error covariance. Adaptive filtering techniques can be introduced to improve adaption under various sensors and scenarios. The filter assumes that all distributions are zero mean Gaussian distributed and the system to be estimated is linear, which is often not the case. However, the filter generally provides satisfactory results regardless. Common variations of the KF, namely the Extended KF (EKF) and Unscented KF (UKF) provide solutions to overcome the linear assumption. Technical explanations are available in [288], [224].

Kalman filters are commonly used for localization, tracking and navigation due to efficiency, robustness and real time processing. Common applications include GPS systems [116], robot positioning [57] and computer vision object tracking [157]. These applications require the ability to quickly combine together complementary noisy data sources from various sensors to provide an improved and controlled response. Even during the loss of sensor data, the KF is capable of estimating the required states completely from the prediction step alone.

Kalman filtering resembles a rudimentary form of Machine Learning (ML), in that it applies new data and weightings to increase its accuracy over time, without being programmed to do so. Much in the same way labelled training data may be applied to a ML algorithm; a system model must be provided to a KF to provide the basis of the prediction. Nevertheless, while Kalman filtering shares similarities to specific ML models, there may be various differences in structure, applications and the type of data. Critically Kalman filtering is state-based or time-based and does not identify patterns or features within data, making it unsuitable for a variety of use cases. However, where appropriate, a KF is considerably less complicated, computationally less intensive and for these reasons, faster. Kalman filtering can be used in conjunction with or in place of more complicated ML techniques [21] to reduce runtime or produce a more general output.

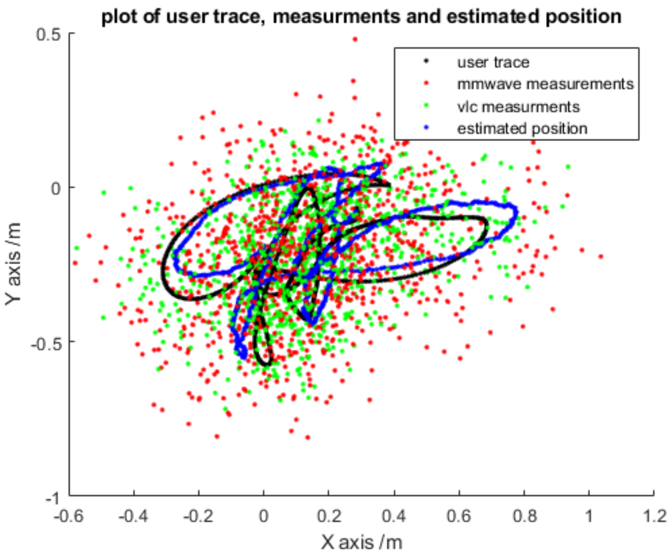


Figure 2-4 Kalman Filter tracking user plot

Figure 2-4 illustrates a top down 2D view of a Kalman filter used to perform data fusion from two independent localization techniques in order to track a virtual reality user in 3D space. The localization techniques are modelled on mmWave TDOA and VLC RSS from the Internet of Radio Light (IoRL) project. The mmWave estimates are shown in red, VLC estimates in green and the ground truth in black. The output of the KF, shown by the blue, is evidently steadier and more accurate than the two noisy estimates independently.

2.4.4 Federated learning

Federated learning is a recent addition to the distributed ML approaches, which aims at training a machine learning or deep learning algorithm across multiple local datasets, contained in decentralized edge devices or servers holding local data samples, without exchanging their data — thus addressing critical issues such as data privacy, data security, and data access rights to heterogeneous data. The Federated learning approach is in contrast to traditional centralized learning techniques where all data samples are forwarded to a centralized server and to classical distributed machine learning techniques, which assume that the local data samples are identically distributed and have the same size. The general design of federated learning involves training local models on local data samples and exchanging parameters (e.g., weights in a DNN) among those local models to generate a global model. Federated learning algorithms can use a centralized server that orchestrates the various steps of the algorithm and serves as a reference clock, or they may be peer-to-peer, where no centralized server exists. The federated learning process is divided into multiple rounds, each consisting of four steps:

- Step 1: Local training - all local servers compute training gradients or parameters and send locally trained model parameters to the central server.
- Step 2: Model aggregating - the central server performs secure aggregation of the uploaded parameters from 'n' local servers without learning any local information.
- Step 3: Parameter broadcasting - the central server broadcasts the aggregated parameters to the 'n' local servers.
- Step 4: Model updating - all local servers update their respective models with the received aggregated parameters and examine updated models' performance. After several local training and

update exchanges between the central server and its associated local servers, it is possible to achieve a global optimal learning model.

Training ML model at the network edge ensures network scalability by distributing the processing from centralized architectures of the Mobile Core/Cloud to the edge located closer to the user. This allows faster response to user requests since computations, data aggregation, and analytics are handled within user proximity. Moreover, it provides latency improvements for real-time applications as ML models are executed near the user. Many 5G applications are characterized by latency stringency and demand; therefore, the latency induced by communicating and executing ML models in the Mobile Core/Cloud may violate these requirements; hence, the edge is preferable for Mobile Network Operators (MNOs).

Federated learning also has some drawbacks, such as heterogeneity of distributed devices, biased datasets, security issues, and coordination of many devices during training, which is highly expensive in terms of communication resources. Regarding its further applicability in future networks, including 5G, the main point in favor of federated learning is the massive number of devices that will be deployed. Thus, combining these devices' data and computing capabilities enables AI in new networking areas. In this context, mobile edge network is a perfect scenario where FL capabilities can gain value [171]. Cyberattack detection [34], [316], base station association [66], [118], and VNF autoscaling [140] are examples of application areas where the advantages of the federated learning approach can be utilized.

2.4.5 Unsupervised learning and clustering methods

Apart from supervised learning, which is a machine learning task of analysing a training dataset and produces an inferred function that can be exploited for mapping new and unseen instances, another valuable type of algorithm is the unsupervised learning.

Unsupervised learning and specifically clustering algorithms are considered as one of the key solutions in order to improve the performance of 5G and beyond networks [199]. In particular, clustering algorithms discover previously undetected patterns in raw data with no pre-existing knowledge and with a minimum of human supervision and divide them in different groups that share common characteristics. More precisely, the clustering algorithms select the relevant attributes for the data analysis to identify the different point of interests and understand the groups of observations. Since the data variables usually vary in range, they need to be standardized prior to clustering. Some clustering algorithms such as K-means and Hierarchical Agglomerative Clustering use Euclidean distance to group the similar data. As a consequence different ranges can cause some problems on the final results and avoid having a variable that dominates the overall solution due to the magnitude. Furthermore, determining the optimal number of clusters in a data set is a fundamental issue, which usually requires the developer to specify the number of clusters k to be generated. Once the value of k is known, the clustering can be performed. Once clusters are identified and in order to extract knowledge from the clustered data, the description of the clusters in terms of the variables used for clustering should be analysed. This process of applying context to the extracted clusters is termed ‘profiling’. Most well-known clustering algorithms are considered the K-Means, Mean-Shift Clustering, Fuzzy C-Means, Hierarchical Clustering, Density-Based Spatial Clustering etc.

Typical examples of 5G-related problems, that could be efficiently addressed using clustering algorithms, are considered the profiling extraction for smart resource orchestration [225], anomaly detection [263], optimization issues in real-world complex network topologies [28] etc. In all those different 5G-related problems there is a lack of prior knowledge in proportion to what should be discovered since the extracted profiles or the types of anomalies that should be detected are not proactively known and this is a fundamental issue that is tackled by Clustering Algorithms for the all the aforementioned explained reasons.

One example of the use of unsupervised learning in 5G network is to minimize the latency in communications by employing a novel fuzzy clustering algorithm. The network model is a fog model which consists of a data plane and a control plane. Within the data plane, the fog computing achieves key objectives by using novel methods such as dense geographical distribution, local resource pooling, latency reduction and backbone bandwidth savings to improve the Quality of Service (QoS). While in the control plane, interference mitigation between multiple devices is coordinated within the fog network. The fog networks are comprised of high power node (HPN) and low power node (LPN).

Figure shows the performance comparison of two methods, the Voronoi tessellation model and a novel fuzzy clustering model. The figure compares the Latency of the proposed algorithm compared to the Voronoi model as a function of bandwidth for K = 8. The proposed model 1 ms latency requirement of 5G applications at 1 GHz bandwidth at 5dB [94].

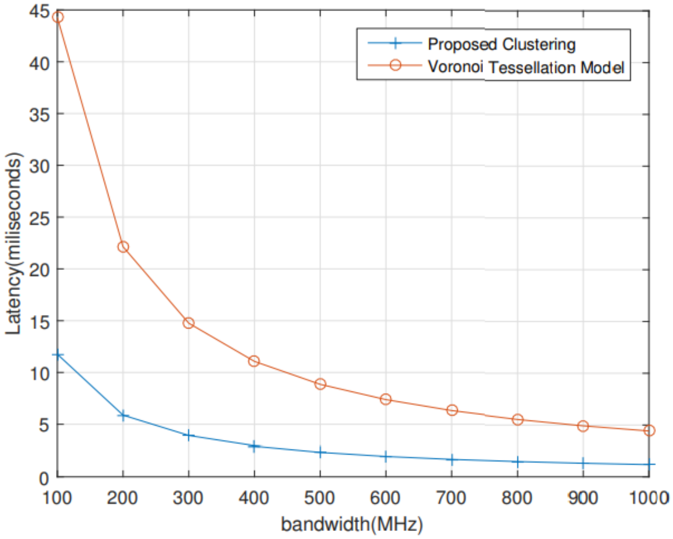


Figure 2-5 Performance example of unsupervised learning - Voronoi tessellation vs. a novel fuzzy clustering model

3 AI/ML for 5G and beyond: use cases

As mentioned in the previous section, AI and ML solutions can be applied in several network domains. In this section we provide a compilation of solutions that are designed, implemented, and tested in the context of 5G PPP projects. More specifically, the following sections describe AI/ML applications to:

- **Network Planning**: Encompassing mechanisms that can guide the planning and dimensioning decisions taken prior to network deployment.
- **Network Diagnostics and Insights:** Including mechanisms used to obtain insights that help operators run the network in a better way (e.g., data traffic forecasts, prediction of failure events etc).
- **Network Optimization and Control:** Characterised by mechanisms that use AI/ML techniques to dynamically reconfigure the network at different time scales. The solutions included in this category are further classified according to their network operational domain, i.e., RAN, transport or compute.

3.1 Network planning

5G, beyond 5G (B5G), and 6G networks will become increasingly complex due to their multi-RAT nature, where parallel layers of connectivity are considered a trend towards disaggregated deployments in which a base station is distributed over a set of separate physical network elements, and the growing number of services and network slices that need to be operated. This growing complexity renders traditional approaches in network planning obsolete, and calls for new automated methods that can use AI/ML to guide planning decisions.

Table 3-1 describes the two network planning use cases that will be described in this section, while providing the interested reader with additional resources related to each use case developed in the corresponding 5GPPP projects.

Table 3-1 Network Planning Use cases

Use Case	5GPPP Project	Additional references
Network element placement problem	ARIADNE	[33], [222]
Application of ML to dimensioning C-RAN clusters	5G-COMPLETE	[232], [233]

3.1.1 Network element placement problem

Future networks will get increasingly complicated due to densification and employment of heterogeneous radio technologies. This leads to large numbers of network elements that make the network deployment very difficult [32]. Regardless of computer aided cellular network design tools, such as 3D-map-based ray tracing tools and field-measurement-based coverage maps, one of the well-recognized problems for radio network design is the network element placement problem [221]. To establish a network of radio transceivers, i.e., base stations or access points (AP) in an area, the network operator needs to identify sites (locations) that would provide the best service in that area. This task is currently approached in a largely manual manner that is highly dependent on a technician’s knowledge and experience, and also requires measurements or estimates regarding the density of user equipment (UE), mobility patterns, demand build-up over time and service coverage, considering that AP(s) are placed in certain locations.

These are network design aspects where ML and AI are seen as potential aids for providing the best possible solutions towards maximum coverage with minimum hardware.

A UE is considered covered if it can connect and receive signals from any AP in a fixed constellation of APs. The identification of the most suitable constellation (i.e., a subset of locations for APs) is a very challenging combinatorial optimization problem where UE-to-AP allocation constraints are to be respected using a minimum number of APs so that AP-related costs (capital, operational, environmental) can be reduced while coverage for UEs is maximized. A feasible solution may also need to respect further resource allocation, scheduling or business constraints. In the literature, this problem is generalized as the location or placement optimization problem, which is a form of set cover optimization problem and is proven non-deterministic polynomial-time (NP)-hard [106], [190].

5G and B5G systems will heavily rely on higher frequencies and densification of the base stations/APs in order to: 1) manage the increased path loss; and 2) increase the single link data rate as well as drastically increase the sum data-rate of the network. Dense network deployment makes the network design more complicated as the range of a single base station is limited. This is where AI and ML can step in to optimize the network element locations and density given a specific propagation environment.

In B5G, we can easily expect thousands of UEs and thousands of APs (including base stations, relay nodes, reconfigurable intelligent surfaces and street-level transceivers) to be placed within a square kilometre area, especially in urban dwellings. Here, AI techniques can play an effective role. Data that is representative of B5G scenarios can be generated by domain experts, to represent network topology comprising of a set of all possible AP locations and a set of UEs. The solution to this problem identifies the near-optimal AP constellation and assignments of UEs to unique APs that result in the best area coverage or network throughput. The placement problem has been faced in other fields as well. In [40], placement problem in chip design is solved by using Reinforcement Learning (RL), while also highlighting challenges posed by RL related to brittleness of gradient updates and the costliness of evaluating rewards. The work also recognizes and cites prior art that used analytical approaches, genetic and hill-climbing methods, integer-linear programming and other heuristics to solve the placement problem.

One particularly important aspect of the dense B5G systems is to manage the cost of the network. As the high frequency systems require vastly more base stations to provide the coverage, also the system becomes more expensive. Albeit the cost for a single small-cell base-station will be cheaper than a full-scale base station site, the total cost of the network, including the hardware, building the infrastructure, and possible rents for the base station sites, can be very high [184]. On top of this come maintenance and electricity costs, leading to high demand for energy efficiency to keep down the costs [210]. These are all facts that also call for high level of optimization in the numbers and locations of the network elements to minimize costs and maximize the network coverage and achievable network throughput.

In the absence of upfront available data, optimization simulations provide a sound mechanism to frame and examine these problems. Some aspects of the network element placement problem can also be individually modelled using ML if past data from similar areas is available. For instance, UE mobility patterns and demand build-up over time can be forecasted using time series analysis. Predictions from these individual models can also be fed as input constraints, to transform the optimizations into predictive optimizations.

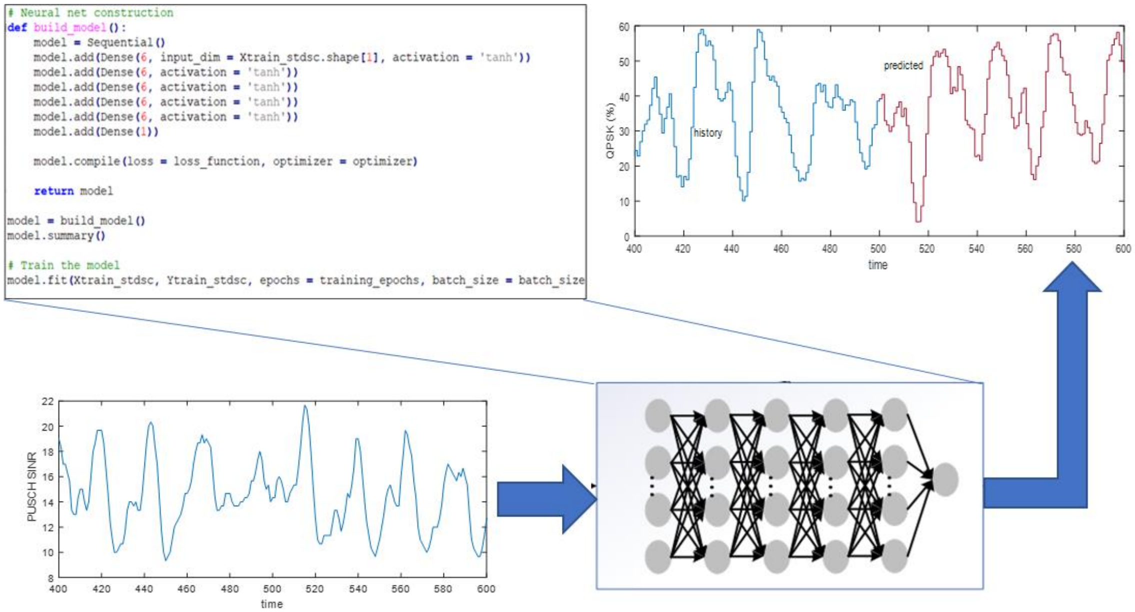
In the light of above, ML and AI play an important role in the future network design. However, there are aspects that still require human intervention. It often happens that the desired base station location is not available due to various reasons, such as lack of electricity or property/land owner refusing to give space for the equipment, among other possible reasons. Thus, the optimal network may not be possible, but ML techniques can be utilized to take into account limitations or revised network designs and adapt a suboptimal solution that maximizes QoS criteria for any propagation environment.

3.1.2 Dimensioning considerations for C-RAN clusters

In C-RAN environments, it is very important to identify optimal allocation of BBU functions to the appropriate servers hosted by the CU, as it is expected to give significant efficiency gains (such as power consumption). Currently, this is performed without taking into consideration the details and specificities of the individual processing functions that BBUs entail. Given that the operation of future C-RAN networks will be supported by virtualized BBU that will operate in a combination of general and specific purpose servers, it is necessary to analyse the specificities and characteristics of the individual processing functions forming the BBU service chain. For example, as discussed in [279] the processing time and CPU utilization of the BBU increase with the channel resources and the Modulation and Coding Scheme (MCS) index. As MCS increases CPU utilization may exceed 80%. Therefore, accurate knowledge of the processing requirements and their evolution over time may significantly assist system operators to design efficient resource-provisioning and allocation scheme in C-RAN environments.

Towards this direction, purposely developed NN models can be used to estimate the BBU processing requirement of individual LTE PHY under various wireless access requirements and traffic load scenarios. Typical examples of NN models that can predict (a) the appropriate PHY layer parameters (i.e., MCS, PRBs, CQI, etc.) and (b) the associate processing requirements of each individual BBU function (i.e. FDMA demodulation, sub-carrier demapper, equalizer and transform decoder, etc.), include the LSTM and MLP NN models.

An example of an MLP-based model that has been developed for the estimation of the percentage of user connections employing a specific modulation format is shown in Figure 3-1. The objective of the MLP model is to predict the modulation scheme distribution of QPSK, accepting as input the PUSCH SINR measurements in a specific time period.



```
# Neural net construction
def build_model():
    model = Sequential()
    model.add(Dense(6, input_dim = Xtrain_stdsc.shape[1], activation = 'tanh'))
    model.add(Dense(6, activation = 'tanh'))
    model.add(Dense(6, activation = 'tanh'))
    model.add(Dense(6, activation = 'tanh'))
    model.add(Dense(6, activation = 'tanh'))
    model.add(Dense(1))

    model.compile(loss = loss_function, optimizer = optimizer)

    return model

model = build_model()
model.summary()

# Train the model
model.fit(Xtrain_stdsc, Ytrain_stdsc, epochs = training_epochs, batch_size = batch_size
```

Figure 3-1 MLP-NN model which has been developed to predict the modulation using as input PUSCH SINR

The same procedure can be applied for other parameters affecting the processing requirements of the BBU [202], enabling advance allocation of BBU functions to the appropriate servers hosted by the CU, giving significant efficiency gains.

3.2 Network diagnostics and insights

Traditionally, operators have relied on expert knowledge to identify problems on a running mobile network. However, the growing complexity of 5G and B5G mobile networks calls for new tools that can autonomously inspect the network state and trigger alarms when necessary. In this chapter we discuss 5G PPP contributions in forecasting and diagnosing techniques in three specific domains:

- AI/ML techniques to forecast network characteristics and events, such as predicting traffic demands or inferring SLA violations.
- AI/ML for high precision user localization, where user location is a critical network insight in several vertical domains.
- AI/ML techniques that can be used to identify and forecast security incidents.

3.2.1 Forecasting network characteristics and events

In this section we discuss the application of AI/ML techniques to forecasting network characteristics and events, including forecasting of traffic distributions in time and space, and forecasting of QoE levels or SLA violations.

Table 3-2 summarizes six use cases in this domain that will be presented in this section, while providing the interested reader with additional resources developed in the corresponding 5G PPP projects.

Table 3-2 Use cases for forecasting network characteristics and events

Use Case	5GPPP Project	Additional references
Synthetising high resolution mobile traffic	5GZORRO	[330], [331]
Efficient mobile traffic forecasting	5GZORRO	[330], [332]
Improving QoS with Forecasting Techniques	5GROWTH	[337], [338], [339]
QoE Inference	5GVINNI	[146]
SLA prediction in multi-tenant environments	5G-CLARITY	[325], [326]
Complex event recognition (CER) & Forecasting	ARIADNE	[242]

3.2.1.1 Synthesising high resolution mobile traffic

Traffic needs in 5G cellular networks vary in each area and time period based on the total demand in terms of users and applications as well as the number of users covered by each mobile cell. Hence, obtaining the needs in terms of bandwidth and taking appropriate response actions to high demands, such as reallocating users to other cells, is quite challenging. The main reasons for this are: 1) the absence of sufficient traffic data, and 2) accuracy in positioning of the users. On top of that, the cost of obtaining fine-grained mobile traffic is high as it requires reports from Mobile Network Operators (MNOs), substantial storage capabilities, and intensive off-line post-processing. In order to simplify the analysis process, MNOs make simple assumptions about the distribution of data traffic across cells. For instance, it is frequently assumed that users and the traffic are uniformly distributed irrespective of the geographical layout of coverage areas. Unfortunately, such approximations are usually highly inaccurate as traffic volumes exhibit considerable disparities between proximate locations. In this section we propose a technique to precisely infer narrowly localized traffic consumption from coarse-grained data recorded by a limited number of monitoring probes (thus reducing deployment costs) that have arbitrary granularity. This is presented in Figure 3-2.

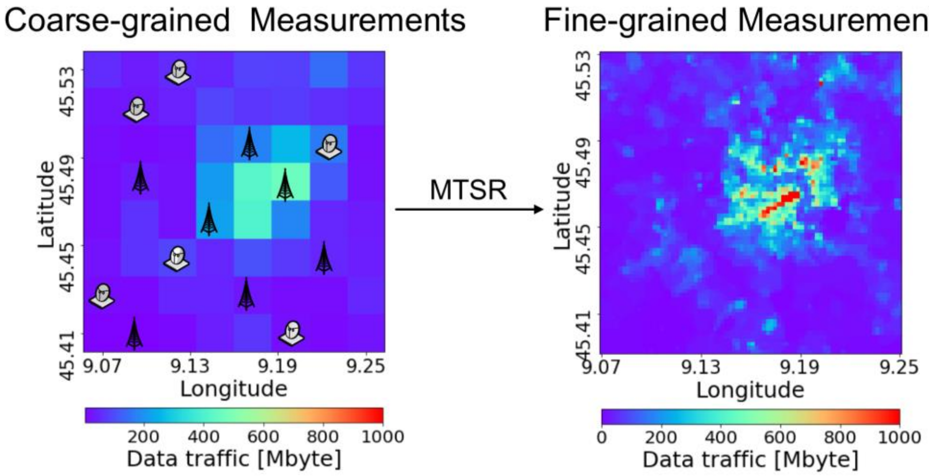


Figure 3-2 From coarse network measurements to high resolution synthetic mobile data traffic [311]

From Figure 3-2 it becomes clear that mobile traffic patterns have spatio-temporal correlation [310] and, as well, can be represented as tensors that highly resemble images (cross-spatial relations) or videos (cross-temporal relations). Moreover, it becomes apparent that a similar problem exists in the image processing field, where images with small number of pixels are enhanced to high-resolution. There, a super-resolution imaging approach mitigates the multiplicity of solutions by constraining the solution space through prior information. This inspires us to employ image processing techniques to learn end-to-end relations between low- and high-resolution mobile traffic snapshots. The ML technique that is employed for obtaining high resolution synthetic mobile data traffic is the Generative Adversarial Networks described in Section 2.4.1. Concerning the deployment, a possible solution would be to consider these techniques at a MEC location which are close to the UE and collect mobile traffic for each cell. This would allow reducing the complexity of the ML models, distributing the processing and providing performance improvements, since only the fine-grained mobile traffic measurements will be communicated to the 5GC in the Cloud. At the same time fine-grained mobile traffic would allow more intelligent resource management for MNOs and would also mitigate traffic congestion in popular hot spots.

3.2.1.2 Efficient mobile traffic forecasting

Network slices are formed using resources and services that are offered by Service Providers (SPs). Moreover, each SP has to provide continuous guarantee that specific Service Level Agreements (SLAs) are met towards the consumers and when degradation in performance is spotted then respective actions are taken to restore the provided services. Providing such guarantees is quite challenging as it requires the deployment of monitoring interfaces for each service to gather accurate and real-time measurements. Then, the gathered measurements are compared against thresholds derived from the SLAs. Thus, a current challenge for SPs is the automated provision of these interfaces that will allow the real-time detection and response to abnormal service operation. This will ensure reliability for the operation of the network slices as well as the associated applications.

Our work based on [282] allows deploying automatically monitoring interfaces for accurately measuring services, including the mobile traffic distribution. These interfaces are deployed upon the provision of a network slice and are applicable at different levels including, a) the cloud/core infrastructure, b) the mobile edge, as well as c) at the User Equipment (UE) level. As an outcome, the monitoring interfaces are 1) collecting measurements and comparing them against thresholds derived from the SLAs, and 2) forecasting potential events and anomalies in the monitored services to allow the prevision of counter measures that would prevent them from happening. However, even if sufficient data are available, forecasting mobile traffic events has a lot of uncertainty, as the further ahead we forecast, the more uncertain we are.

There is growing evidence that important spatio-temporal correlations exist between traffic patterns [310]. These correlations along with contextual information about network bandwidth data are needed to be kept in the memory of the ML model that is used for forecasting. For this reason, we propose the use of dedicated ML techniques, such as the LSTMs mentioned in Section 2.1.3.

The reason for choosing LSTMs is that they are capable of learning long-term dependencies in comparison to standard RNNs. LSTMs can be used to predict network bandwidth fluctuations as well as anomalies before their actual occurrence. These scenarios are comprised of a single series of observations and a model is required to learn from the series of past observations to predict the next value in the sequence. Nevertheless, when applying standard LSTMs in these scenarios a challenge that is faced is that the sequence of contextual information to be stored is large, requiring substantial memory and computation time for training purposes. Hence, although proven powerful when working with sequential information, this model is highly complex and frequently turns over-fitted. A possible solution to this challenge is to replace the inner connections with convolution operations as shown in Figure 3-3.

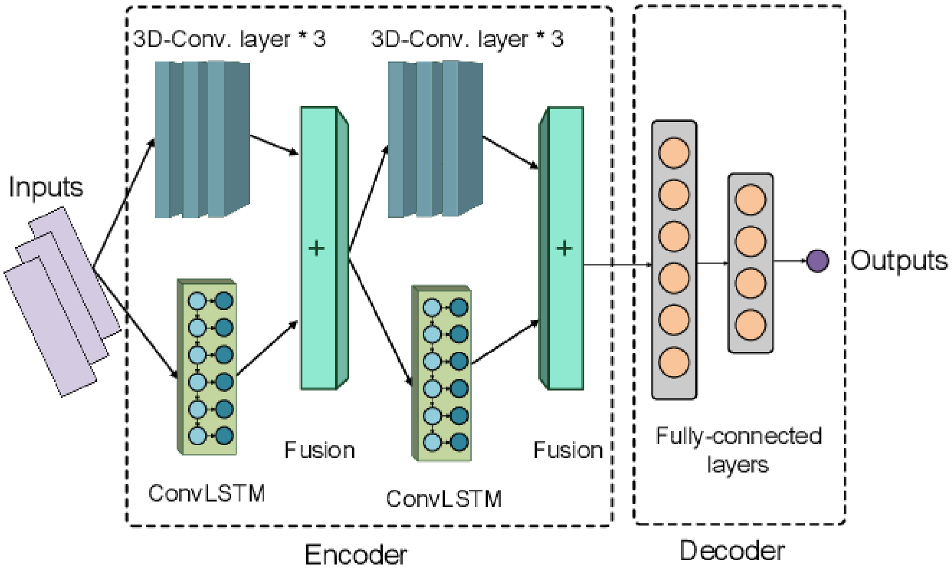


Figure 3-3 Encoder/Decoder architecture used for mobile traffic forecasting [310]

The presented solution reduces significantly the amount of contextual information that should be stored in the forecasting model based on mobile traffic events and enhances the model’s ability to handle the important spatio-temporal information. Additionally, long-term trends present in sequences of data points can be captured, which makes them particularly well-suited for making inferences about mobile data traffic since it exhibits important spatio-temporal correlations [310].

3.2.1.3 Improving QoS with forecasting techniques

Novel 5G use cases typically have strict network requirements (e.g., availability or minimum guaranteed bandwidth). Meeting such requirements is not only a matter of using powerful and recent hardware, but a proper mapping between users' demand and associated resources. However, user demands might vary as a function of time. As an example, the reader can picture a collision avoidance service with MEC servers co-located with radio base stations along a highway. During rush hours the V2N services are very likely to foresee bursts of demand that cannot be accommodated by the turned-on MEC servers along the highway. In this case, a typical approach to dimension resources is to allocate them based on the peak demand. However, such approach might conduct to high inefficiencies, especially if demand variations are high and resources are shared by many services. Forecasting arises as a candidate solution to predict how demand will evolve over time, allowing better and more efficient resource sharing and utilization and, thus, avoiding or minimizing SLA violation by the offered services. For example, auto-scaling, self-healing, and self-reconfiguration mechanisms can be pre-emptively triggered to mitigate the effects that such events could have in the offered services and, in the worst-case scenario, to avoid their downtime. Thus, and at a higher-level, such mechanisms can be leveraged to minimize any violation on agreed SLAs. Forecasting can be implemented based on several algorithms: 1) on classical time series techniques such as Error, Trend, Seasonality forecast (ETS), Auto Regressive Integrated Moving Average (ARIMA), and Exponential Smoothing are the most popular and effective time series predictors; and 2) on AI/ML techniques such as LSTM, 3D-CNNs, and GRUs. Classical time series techniques are fast to train and forecast, however they are neither very accurate nor flexible to adapt to complex data. In contrary, AI/ML-based techniques (e.g., LSTM) can forecast accurately but they require long training. Therefore, selecting the forecasting technique based on the dataset and available resources is crucial.

For example, Figure 3-4 shows how LSTM forecasting technique can be applied to forecast the computational resources of a V2N service. [241] reports the, so called, Enhanced Vehicular Service (EVS), that is a service that deploys sensoring and video streaming and processing facilities in the edge. It reports not only the required physical resources to deploy an EVS service, but as well the flow of cars used to perform their evaluations. The purpose of the traffic flow forecasting is to know whether a deployed V2N has enough resources to meet the E2E delay in an interval of up to 1 hour in the future. Thus, a V2N service can scale accordingly if the 5G network infrastructure receives as input the forecasting information. To relate the number of required resources with the E2E, an M/M/C queuing model is utilized [81].

Three different scaling strategies are considered: (i) *max.scaling (over-provisioning)*: this strategy assumes that the V2N service is deployed with "c" instances capable of meeting the average E2E delay during peak hours of traffic; (ii) *avg. scaling*: the network dimensions the V2N service so that the "c" instances meet latency restrictions considering an average flow of vehicles; (iii) *n-min. scaling*: based on the *n*-minutes ahead forecasting, the service is scaled to satisfy the peak of traffic forecasted for the next "*n"* minutes. In this latter case LSTM is utilised to perform traffic flow prediction.

LSTM is a special form of RNN that can learn long-term dependencies based on the information remembered in previous steps of the learning process. More details about memory blocks, and the multiplication gates are described in the Section 2.1.3.

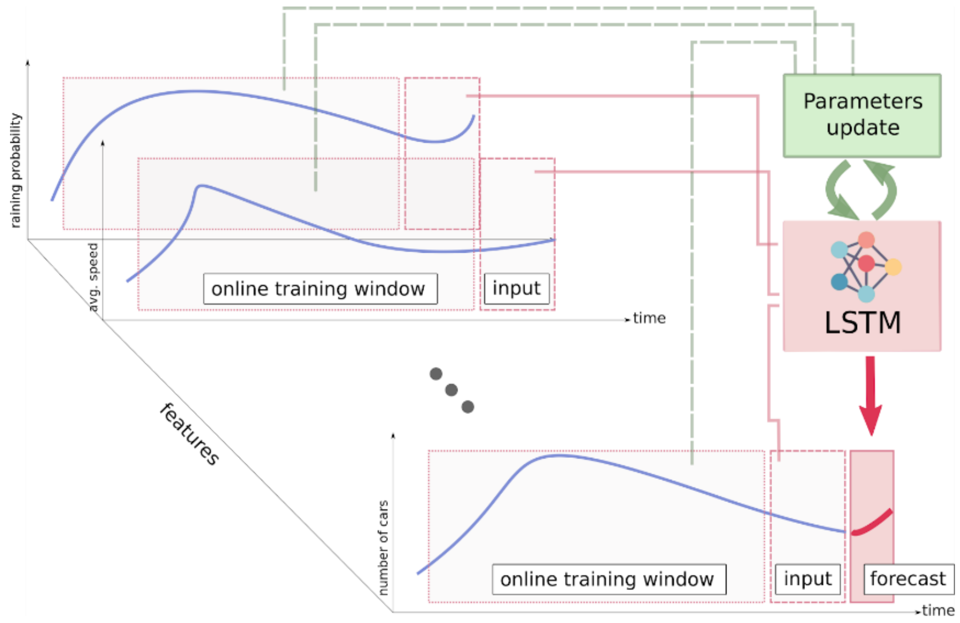


Figure 3-4 LSTM online training and forecasting of Neural Network structure

Figure 3-4 shows how the weights of LSTM are updated in this example: (i) LSTM learns the neuron's weights by running the back-propagation-through-time [298]; (ii) upon each traffic flow forecast, the parameters (e.g., avg. speed) are updated by using an online training window (i.e., the online window incorporates the latest observed features (e.g., avg. speed), and discards the oldest features).

Figure 3-5 (a) shows how the V2N scales the above mentioned strategies. Results show that the n-min scaling is able to reduce the E2E violations, which is the average number of time the E2E delay requirement is violated. The average scaling strategy has more violations during the time period 12:00 to 13:00 and the end of the day (i.e., 16:00 to 20:00). Figure 3-5 (b) further confirms that even with different n-min. scaling is able to reduce violations and Figure 3-5 (c) additionally, and compared to the avg. scaling strategy, n-min. scaling results in the lowest investment increase among the analysed use cases.

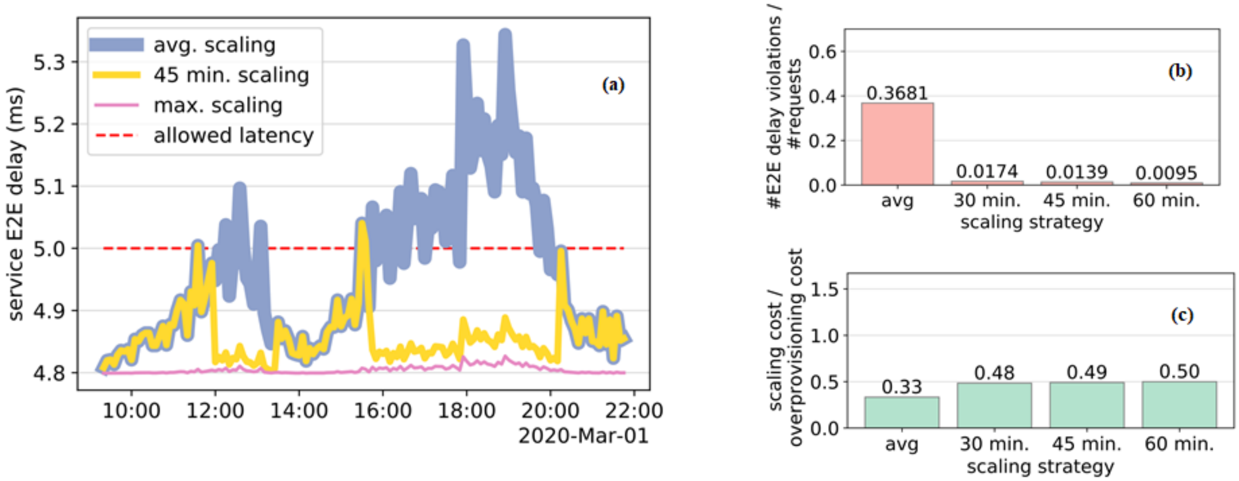


Figure 3-5 Remote driving scaling on (a) service E2E delay; (b) delay violations; (c) savings

3.2.1.4 Quality of experience inference

In a 5G network with NFV and SDN, the physical resources are virtualized and shared across multiple network slices and then customers. As 5G is shifted from network-centric to customer-centric and ultimately aims at assuring QoE, effective infrastructure management should involve QoE and the customers/end-users' perspective. Considering that 5G is provisioning network slicing services to multiple vertical industries simultaneously whose QoE is characterized in different ways, different mappings from their QoE to network QoS and then to infrastructure behaviours are required. AI/ML technologies are a powerful tool to: 1) analyse the complex relationship between QoS and QoE for various customers and end users; 2) predict the behaviour of customers and end users such that service providers and network operators will be aware of potential QoE degradation; 3) recommend optimal solutions for networks to adapt their operations and take proactive actions to assure QoE timely and accurately.

Traditional QoE-QoS mapping models are based on regression models with the objective of finding an explicit relationship between human customer experience and network QoS, and building a mathematical model to control the network QoS for improving the QoE. With the increasing complexity and number of network QoS features, as well as the inclusion of significant context data, more ML mechanisms are used to develop implicit QoE-QoS relationship aiming to identify the key influence factors, e.g., decision trees, recurrent neural network, SVM, Bayesian networks [300]. An outcome of the established QoE-QoS mapping (or QoE-infrastructure KPI mapping) is to predict QoE from network measurements. It is a popular area where general supervised ML classification methods are applied to, such as Naïve Bayesian, SVM, kNN, decision tree, random forest and neural networks. Nowadays with the introduction of NFV and virtualization and abstraction, more variables and features are hidden in the mapping models, which thus obscure the relationship and influence the prediction accuracy. More advanced ML methods are expected to deal with the highly complex and diverse datasets involved in the mapping from vertical QoE to infrastructure KPIs (e.g., with multiple levels of mapping: vertical QoE ←→ application QoE ←→ application KQIs ←→ E2E network service QoS ←→ network domain service QoS ←→ NF KPIs ←→ Infrastructure KPIs), as shown in Figure 3-6 with network slicing.

The linear regression QoS-QoE analysis aims to find an explicit mathematical relationship between QoS and QoE metrics. It is based on curve fitting and mainly applied to the QoS-QoE pairs having quasi-linear relationship. However, the practical QoS-QoE relationship is varying with contexts (e.g., environment, demographic, social status, culture background, etc.), which are not dealt with by the regression models. Decision Tree based models are proposed to interpret the non-linear relationship between QoS and QoE metrics, with consideration of the impact of the external context factors. More importantly, they can handle a large number of QoS metrics simultaneously while the regression models usually are suitable for a small number of QoS metrics. Notice that Decision Tree models only deal with discrete-value QoS metrics, which are not realistic as many network QoS metrics are continuous. Both regression and decision trees can predict QoE but with different granularity. Regression models are able to predict fine-grained QoE whereas decision trees only predict QoE as a range rather than a score [303].

Another direction in the QoE inference is to reduce the number of QoS metrics used to model QoE factors. Principal Component Analysis (PCA) is a typical method to remove the redundancy in the QoS metrics and identify the most influential factors that can be used in regression models to derive an explicit relationship between QoE and the most influential QoS metrics [163]. A similar approach is multidimensional scaling (MDS) that transforms the data into distances between points representing perceptual events in the feature space.

The QoS-QoE mapping is too complex to be solved by a single ML model. Usually, different ML methods are selected to model QoE of different services and applications. For example, support vector machines

(SVM) for web service QoE, recurrent neural network (RNN) for audio and video services, and decision trees for Internet video, and Bayesian network for VoIP applications.

Integrating the QoE prediction model into the network management system is a prerequisite to realize QoE-driven network service optimization. The integrated management system can estimate the QoE based on network monitoring data and reconfigure networks to assure a QoE score predefined in SLA. Some work has been done to integrate QoE ML models with 5G networks, e.g., SDN in [187]. In [187], after evaluating several deep learning (DL) models, the ML QoE predictor for multimedia services is designed to consists of a DL classifier based on a combination of a convolutional neural network (CNN) and a RNN with a final Gaussian process (GP) classifier based on Laplace approximation (Figure 3-7). It not only produces a QoE score but also detects and isolates seven common anomalies that lead to the QoE score. The combination of DL and GP classifier generates the optimal performance. As shown in Figure 3-7, a sequence of two CNN layers extracts new features from 2-D time series of samples and adds these new features to create a new dimension. Then two LSTM layers are inserted to process the flattened time sequence information, creating a final embedding of the data into a 1-D vector, delivered to a fully connected final network for generating the expected prediction (Model 2 or Model 3 in Figure 3-7).

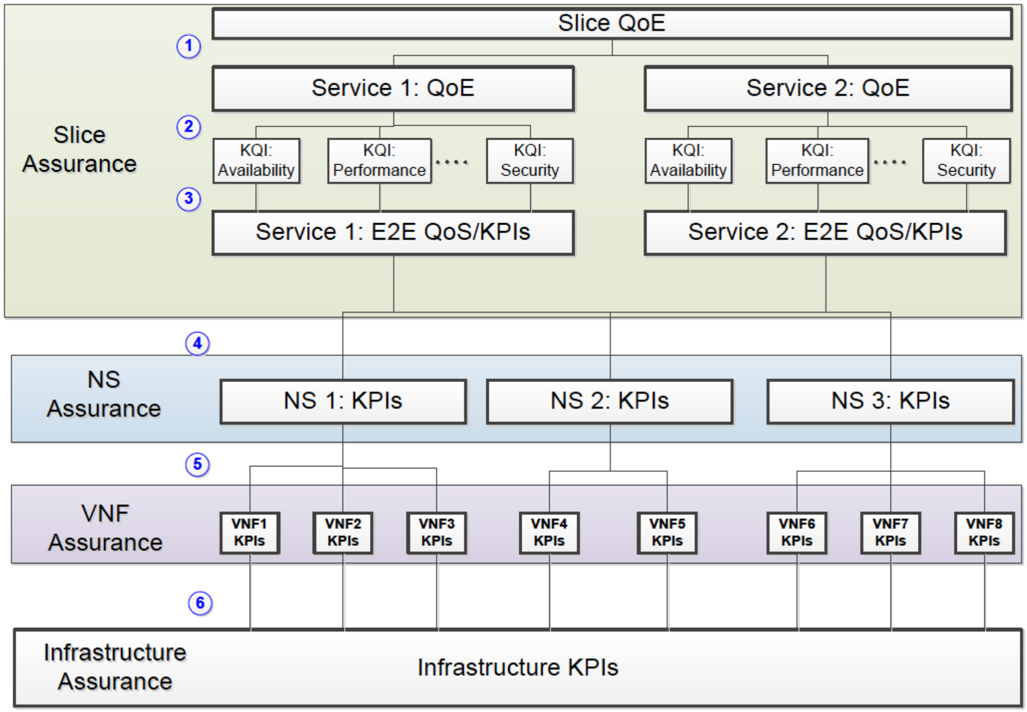


Figure 3-6: Multi-layer QoS-QoE mapping in network slicing [146]

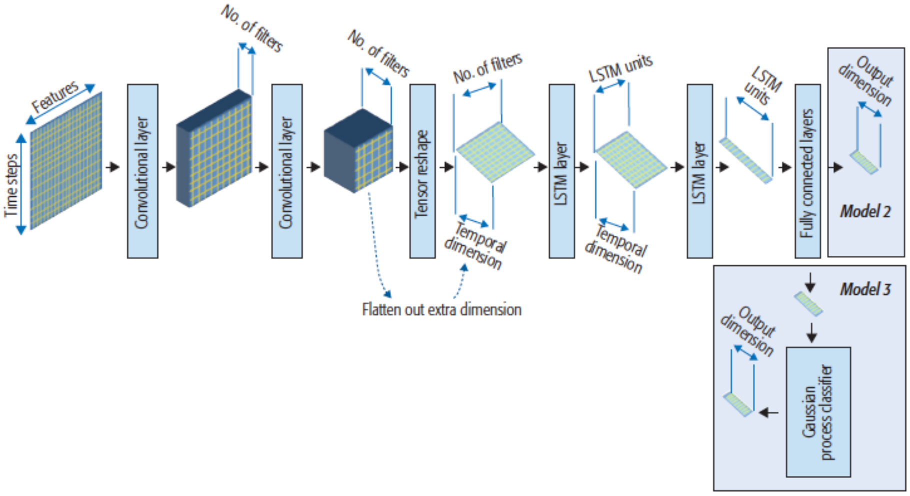


Figure 3-7: Architecture for the best DL network to predict QoE and classify anomalies in [187]

3.2.1.5 SLA prediction with echo state networks

5G network has been designed in a way that efficiently operates multiple virtual sub-networks that sit on the same physical infrastructure and use network slicing approaches to support new services such as ultra-reliable low-latency communication (URLLC), enhanced positioning, massive machine type communication (mMTC), etc. The network slicing approach has also opened various opportunities for MNOs as well as infrastructure owners. One of the disruptive opportunities of the network slicing approach is the “neutral host networks” where an infrastructure owner deploys and manages the network and accommodates MNOs for a specific time/event/service based on some SLAs. As the network infrastructure has limited physical resources, the infrastructure owner (neutral host) should carefully allocate its resources to the MNOs (tenants) to satisfy SLAs. As various service types can be requested by different tenants at the same time, dynamic network characteristics such as user demand, traffic types, spatial load distribution to access nodes and mobility patterns should be forecasted in order to predict possible SLA violations or success rates. A probability margin for possible SLA violations or success rates could then be used by MNO or the infrastructure provider to decide whether to initiate the services/slices or not. At this point, AI/ML intervention is needed to forecast the trend of the network traffic and its spatial distribution and predict the possible SLA violation/success rate. The predicted SLA violation/success rate can be used either for a zero-touch network optimization approach that can automatically initiate other AI/ML models for network control functions (where the high-level description of echo state network (ESN) is depicted in Figure 3-8 [175]) or by MNO/infrastructure providers to interact with the network directly in order to prevent any SLA violations. The further details on the system architecture will be provided in the following section.

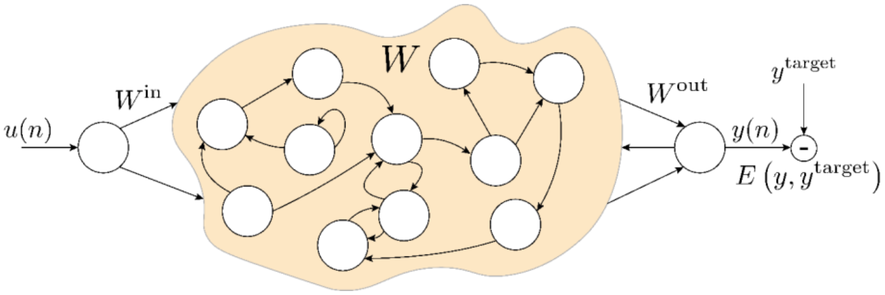


Figure 3-8 A high-level representation of an echo state networks (ESN) architecture

3.2.1.6 Complex event recognition and forecasting

The emergence of 5G networks promises increased throughput, speed, energy efficiency and service reliability in telecommunication applications. A trade-off is that the complexity of such networks may quickly become a bottleneck in analysing their behaviour, troubleshooting suboptimal functionality and optimizing their performance. Due to the temporal/dynamic nature of 5G applications, which makes them event-based to a large extent, the ability to timely detect, or even forecast events of special significance for the "life" of such networks is of utmost importance for dealing with their complexity in a principled manner and extracting valuable insights on their behaviour. Monitoring and reasoning with such events and their spatio-temporal patterns, related e.g. to network resource management and connectivity, network congestion, suboptimal operation or failure, to name but a few, may greatly facilitate decision making and foster proactivity towards the optimization of the network's performance.

To address such issues complex event recognition (CER) & forecasting techniques can be utilised. CER systems [73], [111], [39] are concerned with detecting event occurrences of interest in heterogeneous, correlated, high volume/velocity streaming input, e.g. data generated by monitoring the evolution of a large mobile network in time. The target events that are to be detected are often called complex events, and are defined as spatio-temporal combinations of input time-stamped pieces of information (often called simple events) in predefined patterns. The detected complex events may be utilized by decision makers (either human, or algorithmic), in order to take appropriate actions towards e.g. preventing undesired situations, optimizing aspects of the network's performance and so on.

CER applications face several challenges, all of which are present in the telecommunications domain. The massive data volumes of 5G applications call for scalable event processing methodologies that allow for high input event throughput and low latency in delivering the recognition results. Dealing with the ubiquitous noise in the data collected/generated as a network functions requires noise and uncertainty-resilient event pattern matching techniques. Also, the dynamic nature of the domain calls for adaptive CER strategies, capable of utilizing machine learning tools to update the knowledge base of complex event patterns to reflect change in the characteristics of the input data, or even discover novel patterns from scratch. Finally, Valuable existing expert knowledge about the domain may greatly facilitate learning & reasoning for CER if taken into account. This calls for expressive, yet highly efficient form an operational perspective, event pattern specification languages that allow to easily encode domain principles into usable background knowledge for reasoning & learning.

These challenges are being addressed by building on CER approaches based on computational logic as a unifying representation language for input events, complex event patterns (represented by temporal logical rules) and background knowledge. This approach is supported by a highly efficient temporal reasoning engine [44], optimized towards the needs of high-throughput CER applications and capable of scaling up to very large data volumes and high velocity data streams. Uncertainty handling and machine learning/revising complex event patterns [152], [151] are supported via Statistical Relational AI techniques that combine logic with probability and machine learning [252].

An event pattern can either be fully matched against the streaming data, in which case events are detected, or partially matched, which allows for future occurrences of events to be forecast with various degrees of certainty. The latter usually stems from stochastic models of future behaviour, embedded into the event processing loop, which project into the future the sequence of events that resulted to a partial event pattern match, to estimate the likelihood of a full match, i.e. the actual occurrence of a particular complex event. Complex Event Forecasting (CEF) takes CER one step forward from and may be a key enabler of proactive decision-making in complex networks.

Notably, there is a conceptual difference between forecasting and prediction, as the latter term is understood in machine learning, where the goal is to "predict"' the output of a function on previously unseen input data, even if there is no temporal dimension. In CEF time is a crucial component and the goal is to predict the future output of some function or the occurrence of an event. Time-series forecasting is an example of the former case and is a field with a significant history of contributions. However, its methods cannot be directly transferred to CEF, since it handles streams of (mostly) real-valued variables and deals with relatively simple patterns. On the contrary, in CEF we are also interested in categorical values, related through complex patterns and involving multiple variables, and the goal is to forecast the occurrence of any type of situation that may be defined as an event (e.g. a network congestion event).

CEF is supported in [242] by an approach based on Pattern Markov Chains [38]. Complex event patterns are represented by automata structures defining relations between input events. During an initial training period the system consumes a portion of the input stream and encodes regularities therein in event occurrences related to the complex event pattern of interest. Subsequently, as new events arrive, the CEF system is able to output future time intervals (and associated probability values) in which the pattern is likely to be fully matched, given the events that have been observed so far. As more events are consumed, the system revises its forecasts to reflect possible changes in the state of the pattern. Essentially, the aforementioned initial training period learns a probabilistic model for the complex event pattern, with which forecasts with guaranteed precision may be produced, in the form of intervals within which a full match is expected.

3.2.2 Estimating user locations

Accurate user positioning is of paramount importance for industry verticals such as Industry 4.0, where real-time monitoring of assets and robots is critical to the overall business efficiency. User positioning is already included in 5G standards, where it targets meter level accuracies, still far from the cm-level accuracies required in some domains. User positioning is a problem well-suited for the application of AI/ML techniques that can fuse positioning data from different technologies, or aid in determining the line-of-sight path in a multi-path propagation environment.

Table 3-3 summarizes three applications of AI/ML to positioning use cases described in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-3 Use cases for estimating user location

Use Case	5GPPP Project	Additional references
AI assisted sensor fusion	5GSOLUTIONS	[319]
5G Localization based on Soft Information	LOCUS	[22], [259]
5G Localization based on Sequential Autoencoding	LOCUS	[340]
ML assisted LoS/NLoS discrimination	5G-CLARITY	[325], [326]

3.2.2.1 AI assisted sensor fusion

5G technology promises to achieve unprecedented positioning precision thanks to the favourable radio attributes that allow to reach sub 1-meter accuracy. In fact, the use of centimetre and millimetre bands (cmWave and mmWave) where ample spectrum is available makes it possible for technologies such as Ultra-Wideband (UWB) to be leveraged yielding excellent positioning performance. Such Radio Access

Technology (RAT) based positioning can avail of several techniques to accurately determine the position of a User Equipment (UE). The following is a non-exhaustive list of these techniques [3]:

- Enhanced Cell ID (E-CID) like techniques
- Received reference signal power-based techniques
- Carrier-phase based techniques
- Angle based techniques such as Angle of Departure and Arrival (AoD/AoA)
- Timing based techniques such as Time Difference of Arrival (TDoA) and Round Trip Time (RTT)

A combination of part/all the above techniques applied at higher frequency radios has the potential to tremendously improve the accuracy of positioning. However, there are still some limitations in terms of the applicability of such solutions in some specific scenarios. In fact, higher frequency bands suffer from steep performance degradation in low SNR scenarios compared to lower frequency bands. A low SNR can be caused by a multitude of factors such as high noise levels, interference levels, or absence of a Line of Sight (LoS) as shortwave radio signals lack the ability to penetrate solid obstacles such as cars, walls and furniture. In other words, RAT-only dependent solutions may not reach the desired results in scenarios like indoor and dense urban areas. This is why other positioning techniques can be seen as an alternative, which combine RAT dependent with AI assisted positioning. Such hybrid solutions [302] have the potential to bridge the gap and enable accurate 5G positioning where RAT-only solutions would be short of meeting the sub-meter accuracy requirements.

One such solution relies on multimodal data to track the movement of UEs using motion sensors, like the accelerometer and the gyroscope, then map it to the RAT based localization as well as the map information, in order to narrow down the range of possible locations. This data can further be augmented by the trajectory information (speed and direction) of the moving UE which can be inferred from the motion information. For example, a UE that is moving in a corridor is restricted by the buildings around it. Motion information can indicate that the UE is moving in a given direction, which narrows down the possible position of the UE with regards to the radius obtained using RAT-only information, as shown in Figure 3-9.

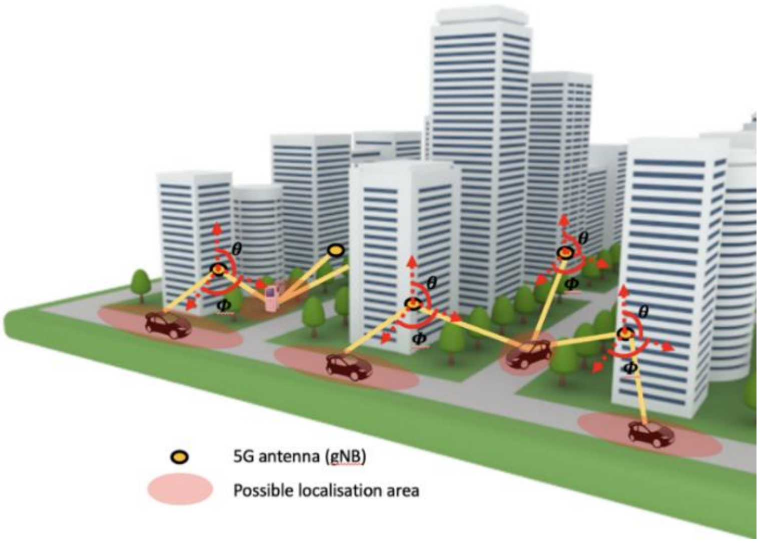


Figure 3-9 illustration of range reduction of the localization combining RAT-based positioning, map and UE's motion information

A ML model can be developed to take as an input the radio parameters, the RAT-only based positioning, the map information, as well as the motion information of the UE. The model can be trained to output a more accurate localization than the one calculated using the RAT-only methods.

Solutions such as the one described above have obvious benefits as they can only improve the accuracy of the 5G-based positioning. However, the presented solution only works in scenarios where the UE is in motion and the benefit may be voided in situations where the UE being tracked is stationary. To tackle this, another solution that stands out is video-based positioning where RAT dependent positioning is assisted by high quality video footage. This is enabled using object recognition and detection with CNNs. In this approach, Perspective-n-Point (PnP) technique [167] can be used to project objects detected in 2D images using a set of n 3D points in space. This allows objects to be positioned relative to other points in the space. In case of 5G positioning, the RAT based position is augmented by the relative location of the UE to other points in the room. This allows for more precise location information to be extracted. Here also, AI is used to cross check and reduce the range of the possible locations provided by the RAT dependent calculation, which can only improve the accuracy.

Another option is to use 5G radio attributes to extend the scope of positioning beyond UEs and connected objects. Device-free localization is a technique than can be explored in 5G in which backscattered radio signals can be used for passive tracking of objects. The above AI-based solutions can also benefit device-free localization as these can be combined with any RAT dependent positioning solution.

To conclude, 5G-based positioning is a great example of where injecting AI can only yield in a positive sum as new context information is leveraged to increase the positioning accuracy. Data fusion techniques are investigated to showcase the added value of such hybrid techniques in scenarios when 5G RAT-based localisation alone may fall short of achieving the required positioning accuracy [240].

3.2.2.2 5G localization based on soft information

Conventional 5G localization relies on single-value estimates (SVEs) based on 3GPP defined signal features such as uplink (UL) time difference of arrival (TDoA), downlink (DL) TDoA, received signal strength indicator (RSSI), angle of arrival (AoA), or angle of departure (AoD), which serve as inputs to a localization algorithm for position inference. This type of localization suffers in harsh wireless environments, where multipath, shadowing, and NLOS conditions impair significantly the measurements and the quality of such estimates.

In [22], Soft Information is proposed as a basis for positioning to overcome the limitations of SVE-based localization, and to leverage radio information available in different radio channels. The Soft Information (SI) encapsulates all the information available from measurements and contextual data at the UE at a given position. Such information could be sensing measurements (e.g., using radio signals), digital map, UE profile, etc. from which generative models for the likelihood of location-related features given measurements are learned from the environment.

In [258], using unsupervised ML techniques, statistical characterizations of the relationship between measurements and ranges, namely soft range information, is obtained from range-related measurements and then those statistical characterizations are used to determine the UE positions.

SI is composed of soft feature information (i.e., the ensemble of positional information associated with measurements) and of soft context information (i.e., the ensemble of environmental information associated with contextual data), and it can be determined by a two-phase algorithm summarized below:

- off-line phase where the approximate of the generative model is learned in a Bayesian setting based on a joint distribution function from measurements, positional features, and context data; and,
- on-line phase where the soft feature information and soft context information for each new measurement are determined based on the generative model learned in the previous phase.

Figure 3-10 shows the ECDF of the horizontal localization error for Indoor Open Office (IOO) scenario with Positioning Reference Signal (PRS) bandwidth of 100 MHz bandwidth at 4 GHz, where the blue dots represent a set of results presented in [3] under similar simulation settings, dashed red line represents results obtained using SVE-based localization algorithms, and solid green line represents the results obtained with SI-based localization algorithms. It can be observed that SI-based approach provides a noticeable performance improvement compared to the SVE-based approach, especially in the tail of the ECDF curve. At the 90-th percentile, SI-based localization shows approximatively 3 m of horizontal localization error and an improvement of 6 m compared to SVE-based localization.

Performance of SI based positioning can be enhanced by further fusion of more measurements, and support for more bands, which is a natural integration in such a framework.

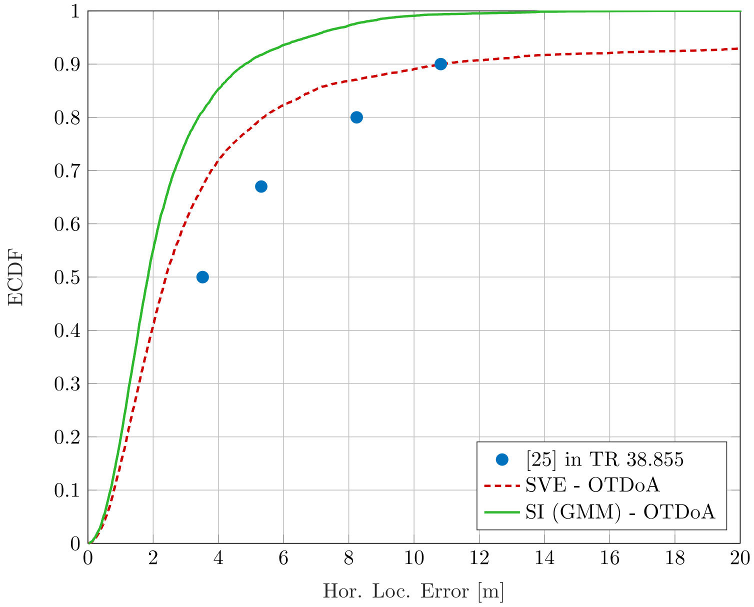


Figure 3-10 ECDF for the horizontal localization error: IOO scenario with PRS bandwidth of 50 MHz at 2.5 GHz

3.2.2.3 5G localization based on sequential auto-encoding

The work presented here investigates fingerprinting for positioning, it is motivated by similar concerns as Section 3.2.2.2, regarding harsh radio environments that hinder the availability of high-quality measurements relied upon by GNSS and single-value estimation based positioning techniques. The work in the section has the following aspects:

- Leverages temporal and sequential aspects of RF signals in fingerprint generation.
- Leverages standardised measurements, and existing or practical and easy to support mechanism to make signal properties available for fingerprinting.
- Provides a flexible positioning solution that can complement other existing solutions, and opportunistically leverage them.

The approach followed is to train a sequential autoencoder model capable of estimating position from signal measurements (RSRP and RSRQ). The autoencoder learns to generate a lower dimensional representation of sequence of input signal measurements, then estimate UE position based on the generated representation. In this sense, the trained model, when deployed, will be able to map signals RSRP and RSRQ values to a latent representation (fingerprint) then use the latter for position prediction.

The model we built is recurrent neural network (RNN) autoencoder-based. The autoencoder is composed of and encoder and decoder, that consist of layers of stacked RNN units. In the forward pass of the training process, the encoder receives the input data in the form of fixed length ordered sequences of measurements, and outputs a fixed size vector that ultimately will converge to be a “good” latent representation (fingerprint). The decoder receives context information from the last layer of the encoder, and ultimately learns to reconstruct the coordinates corresponding to the input sequence.

Results obtained from simulating 150 UEs performing indoor mobility according to the Waypoint Random Mobility Model, under the coverage of 14 FemtoCells, yield a mean positioning error of ~2 meters and less than 3 meters for 86% of position estimates using our model.

Sequential autoencoding can be fine-tuned for better performance by searching the parameter space of the encoder-decoder networks (Length of input sequences, depth/width of the model, learning rate…etc). The effect of initial position labelling needs to be assessed with regards to the accuracy provided by available techniques, in addition to the effect of measurements noise and UE capabilities. This observation will form the basis for further investigation of this work.

3.2.2.4 ML assisted LoS/NLoS discrimination

Non-Line-of-Sight (NLoS) identification is key for precise location estimation in time-based localization algorithms. In particular, NLoS links introduce a positive bias when estimating the position of a user. Therefore, identifying whether a communication link between an AP and a user is NLoS or LoS is of crucial importance if the user is to be positioned with high accuracy. Following the identification, the detrimental effect of NLoS links need to be mitigated to achieve accurate positioning.

Among all the existing NLoS identification and mitigation algorithms, ML-based approaches have drawn a major attention thanks to their superb performance and low complexity. As a successful candidate, Support Vector Machine (SVM) [305], or its variants such as Least Square SVM (LS-SVM) or Relevance Vector Machine (RVM), has been proven to outperform others [307], [277]. In particular, SVM separates the classes (in this case only two, namely LoS or NLoS) by a gap whose width is as large as possible. For the input data, which requires non-linear separation, one can employ *kernel*s, enabling the transformation of the input data to higher dimensional feature spaces where a linear separation is possible. Mathematically, SVM for data classification can be described by

$$c(\mathbf{x}) = \mathrm{sgn}[\mathbf{w}^T \varphi(\mathbf{x}) + \mathbf{b}],$$

where $c(\mathbf{x})$ calculates the class (1 or -1) given the feature vector $\mathbf{x}$, $\varphi(\mathbf{x})$ is the non-linear transformation, and sgn[.] denotes the a function whose value is 1 for $x>0$, -1 for $x<0$, and 0 for $x=0$. Vector parameters $\mathbf{w}$ and $\mathbf{b}$ are the parameters to be learned from the training data. For the above-mentioned function we extract the feature vector $\mathbf{x}$ from Received Signal Strength (RSS) measurements, which can be measured almost in all APs or, alternatively, from the Channel Impulse Response (CIR), which is featured in most of the new AP devices. Typical features are energy of the received signal, Maximum amplitude of the received signal, Mean excess delay, RMS delay spread, and Kurtosis [257].

Having the status of the communication link identified (NLoS or LoS), one can rely on SVM regression to mitigate its negative impact. The regression function is obtained by

$$d(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) + \mathbf{b},$$

where d(**x**) denotes the distance corresponding to the communication link. In fact, in the regression we are trying to find the function which maps the feature vector **x** to the AP-user distance d**.**

The main difference between the variants of SVM is how one solves the optimization problem for obtaining the parameters **w** and **b** or, alternatively, the learning process. For example, apart from the slightly different objective function, SVM requires solving a quadratic programming problem while for LS-SVM a linear system is solved. RVM, on the other hand, employs a Bayesian learning process to obtain the abovementioned parameters, albeit with far less number of (relevance) vectors [182].

3.2.3 Forecasting security incidents

Future networks are incorporating new technologies, such as SDN and NFV, which however give raise to new security threats, requiring new security solutions. In this sense, the use of ML and DL techniques is gaining more importance in the last years within cybersecurity research. Modern attacks [168], [51], [35] being addressed in this direction are:

- **Device-centred attacks.** These attacks vary depending on the purpose or objective of the attacker. We have *identification attacks*, whose objective is to discover device hardware and software characteristics to gain information from the network environment, or uniquely identify each one of the present devices, highly affecting network privacy. There are also *Bidding down attacks*, which degrade performance of a device by degrading it to older networks such as 2G or 3G. Besides, some attacks are centred on *Battery draining*, targeting resource constrained IoT devices with the objective of making them inoperative.
- **Base station attacks.** In this category, attacks affect to the network access points, preventing service to users or enabling more advanced attacks. Examples of this type of attacks include *bandwidth spoofing attack*, where fake APs use the same frequencies and identifiers that a legitimate one to perform *Man-in-the-middle* and *Eavesdropping*, or *Denial of Service (DoS)* performed using mobile botnets or j*amming techniques.*
- **Attacks on multi-tenant network slices**. In contrast to previous generations, 5G networks include multi-tenant networks addressed through network slicing. These network slices present a new attack vector to perform network attacks. *DDoS flooding attacks* in this scenario can cause service disruption in the entire slice, affecting even to slice-shared physical link and core network components, impacting the proper performance of other slices. DDoS attacks are already common in current networks but attacks directly related to slice management have also emerged, such as *Slice-initiated attacks*, which focus on the modification of the VNF/slice configuration to exhaust hardware resources, or *side channel attacks*, which focus on data leakage by performing information gathering from other slices running in shared hardware.
- **Vulnerabilities in Firmware, Software and Protocols.** The explosion in the number of services offered brings with it an exponential increase in the software protocols to be developed. Thus, the detection of vulnerabilities and their correction before they are exploited is a key aspect of the networks of the future. In this area, one of the key points is the maintenance of service security over time, not leaving vulnerable versions operational.
- **Traditional network attacks.** Network attacks common in earlier networks are still present in modern networks, and are even enhanced by the increased number of devices and improved network performance. Then, the methods for detecting and mitigating common network attacks, such as *massive horizontal and vertical port scanning, botnets, service DoS/DDoS or ransomware*, need to be improved according to the evolution of the attacks themselves.

The 5GPPP community is investigating how AI/ML techniques can be used to detect some of the previous attacks. Table 3-4 summarizes a set of security forecasting use cases discussed in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-4 Use cases for security forecasting

Use Case	5GPPP Project	Additional references
Network Traffic Inspection	5GZORRO	[185], [330]
Real-time detection of DDoS attacks	INSPIRE-5GPLUS	[341]

3.2.3.1 Network traffic inspection

Based on the monitoring of the entire network, different ML/DL models could be created for each category or type of attack to be detected and mitigated [54]. Some proposals, like [207], suggest the application of Deep Reinforcement techniques to improve network security and forecast possible issues. Enabled by the flexibility of this approach, as the objective function can be personalized for different environments, these techniques are being applied to cover host- and network-based intrusion detection, jamming detection, spoofing detection, malware detection, or SDN security, among others.

For network traffic inspection, a common approach is to process traffic as a temporal series [185], either applying window-based statistics to generate vectors used as input in traditional methods such as MLPs, or directly applying RNNs or one-dimensional CNNs as they are the most common techniques for time series processing. These solutions are usually applied for *port scanning*, *botnet*, and *DoS/DDoS* detection and mitigation both using supervised and unsupervised approaches. Moreover, Federated Learning is also employed for similar purposes in 5G networks [174], but moving the data processing into the edge of the network.

Clustering techniques are usually leveraged for security incident forecasting. In this sense, clustering is applied for network activity monitoring, grouping similar usual behaviours together. Besides, it can also be used for threat comparison and analysis.

Note that these examples are just some of the applications found on the literature for security incidents detection and forecasting, as ML/DL techniques can be applied in very variated perspectives according to the objective to cover.

3.2.3.2 Real-time detection of DDoS attacks

This section describes an AI-based anomaly detection system for real-time detection of different kinds of DDoS attacks, perpetrated not only over network-level 5G traffic, but also sophisticated DoS attacks (such as slow DoS attacks done over application-level encrypted traffic) which are difficult to be identified. This system is being defined and implemented in [244], and can monitor in real time the network traffic, analysing, processing and aggregating packets into conversation flows, getting valuable features and statistics that are dynamically analysed in streaming for AI-based anomaly detection.

The system is being designed in a modular way to effectively differentiate the functionalities that are present on it. Firstly, we will dispose a monitoring module that will be in charge of monitoring network traffic, extracting the relevant information from each intercepted packet and arranging this information to the second module, that will group the raw packets into conversations. We consider two different packets to belong to the same conversation if the source and destination IP address pair, as well as the ports, are identical. From each conversation, this second module must be able to calculate a set of representative features. Once the traffic has been grouped into conversations and a list of features for each conversation has been generated, the third and final module will be responsible for detecting the attack using AI techniques. This module is the most important part of the system, as it is responsible for obtaining the conclusion of whether the traffic captured for a conversation, based on the metrics obtained, is an attack or not. The design of each one of the proposed modules will be made so that most of the execution time is spent in the last module, related to the attack detection itself. The first two modules will be designed to be as efficient as possible, making use of streaming-processing techniques and programming languages suitable for this.

The AI-based system runs in a fully distributed way to achieve scalable and efficient detection, and might combine different AI techniques, such as clustering analysis for anomalous detection along with deep learning techniques, in order to increase detection accuracy in those cases where clustering obtains ambiguous probabilities. These proposed models are previously trained using genuine traffic, which will later allow the system to detect anomalous patterns that will identify as attacks. The design and deployment of our system allows both monitoring and final detection of the attack to be done in real-time. The monitoring module is constantly analysing the network traffic, and the detection module will be able, once the conclusion has been reached as to whether the traffic belonging to a conversation is an attack, to carry out mitigation actions in order to dissipate the attack, without having to stop the system, which will continue to analyse the new traffic in order to carry out the same procedure.

3.3 Network optimization and control

The application of AI/ML techniques to network optimization and control is the ultimate goal behind the introduction of AI/ML in networking, where AI/ML functions act on the network, rather than only assisting in network planning or in forecasting events.

AI/ML based network optimization and control is the most challenging application of AI/ML in mobile network and is therefore widely investigated within the 5GPPP community. In this section, we introduce AI/ML use cases for network optimization and control classified according to the network domain where they apply, namely: i) RAN, ii) Transport, iii) NFV infrastructures, iv) E2E network slices, v) Security, and vi) the Application Function domain.

3.3.1 Radio access network

We start discussing the application of AI/ML in the Radio Access Network (RAN) domain. To classify the proposals produced by the 5GPPP community in this domain we take as reference the AI/ML control loops introduced by the O-RAN alliance, which are described in detail in section 4.3.3. These control loops are:

- non real-time: AI/ML techniques act on the network on time scales above 0.5 second.
- near real-time: AI/ML techniques act on the network on time scales between 10 ms and 500 ms.
- real-time: AI/ML techniques act on the network on time scales below 10 ms.

3.3.1.1 Non real-time use cases (>0.5 sec)

Non real-time use cases consider the application of AI/ML techniques closer to the management plane. These techniques can be readily applied to 5G networks where programmability is enabled by SDN and NFV.

Table 3-5 summarizes three RAN non real-time use cases presented in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-5 Use cases for non-real-time RAN aspects

Use Case	5GPPP Project	Additional references
RAN slicing in multi-tenant networks	5G-CLARITY	[325], [326]
Radio Resource Provisioning in multi-technology RAN	5G-CLARITY	[327], [326]
RL-empowered User Association	5G-HEART	[181], [324]

3.3.1.1.1 RAN slicing in multi-tenant networks

A multi-tenancy scenario is the one with high density of people in which multiple operators offer their services on the top of the same physical infrastructure sharing the radio resources (e.g., stadiums, museums, shopping malls, etc.). People moving along the venue requesting some kind of service such as eMBB mostly characterizes these network scenarios. The infrastructure sharing can be possible with the Multi-Operator Core Network (MOCN) solution, where an operator owing the next-generation RAN (NG-RAN) can offer it to other operators in order to deploy their services. Such operators will play the role of tenants. Network slicing can also be used to provision different virtual networks (or slices) per tenant with different characteristics using the same physical infrastructure.

RAN slicing in multi-tenant networks is one of the proposed ML algorithms, which focuses on the multi-tenancy scenario [230]. This algorithm considers a private venue owner of a NG-RAN infrastructure composed of N cells. The physical infrastructure is shared among different tenants, each of them provided with a RAN Slice Instance (RSI). The different tenants can be, for instance, different Mobile Network Operators (MNOs) that provide service to their own users through the private network following a neutral host model. The considered problem consists in determining how the available capacity is distributed among the different RAN slices in the different cells. The capacity share of tenant *k* at time step *t* is defined as $\alpha_t(k)=[\alpha_t\ (k,1),\dots,\ \alpha_t(k,n),\ \dots,\ \alpha_t(k,N)]$, where each component $\alpha_t(k,n)$ is the resource quota assigned to tenant *k* in cell *n* given by the proportion of the total physical resources $N_T(n)$ in the cell provided to the tenant during time step *t*. Given the dynamics of the traffic, a smart capacity sharing strategy will be proposed, which will dynamically determine the resource quota allocated to each RAN slice in each cell and configure the network accordingly. The objectives of the capacity sharing approach are the achievement of an efficient utilization of the available resources and the fulfilment of the SLA established between the private venue network owner and each tenant. With the focus on dealing with the complexity of the computation of the capacity share of each tenant at every time step in a multi-cell scenario, the solution is designed as a multi-agent reinforcement learning (MARL) where each RL agent is associated to a tenant that learns the policy $\pi(k)$ to tune the capacity share dynamically by interacting with the environment. Due to the necessity of the continuous learning from the environment and the expected large state and actions spaces, each agent derives its policy according to a DQN (Deep Q-Network) based algorithm as the RL method, which is able to combine RL with a class of artificial neural network known as DNNs. DQN algorithm follows Bellman´s equation, as indicated below:

$$Q(s,a) = E\left[r + \gamma \max_{a'} Q(s', a')|s, a\right]$$

At each time step *t*, each agent obtains the state $s_t(k)$ from the environment and, based on the policy $\pi(k)$, triggers an action $a_t(k)$ to tune $\alpha_t(k)$. Moreover, a reward signal $r_t(k)$ that reflects the obtained performance after the last action $a_{t-1}(k)$ is provided to the *k*-th agent. The definition of these parameters are as follows:

- The state of tenant k at time t is denoted as $s_t(k)$ = [s_t $(k,1),\ldots,$ $s_t(k,n),$ $\ldots,$ $s_t(k,N)$], where element $s_t(k,n)$ corresponds to cell n and is defined by the triple <$\rho_t(k,n)$, $\alpha_t(k,n)$, $\alpha_{ava,t}(n)$>, where $\rho_t(k,n)$ is the fraction of physical resources used for data traffic by tenant k in cell n in time *t and* $\alpha_{ava,t}(n)$ is the available capacity share in the cell not assigned to any tenant.
- The action $a_t(k)$ = [$a_t(k,1),\ldots,$ $a_t(k,n),$ $\ldots,$ $a_t(k,N)$] for all the cells composed of the cell-specific actions $a_t(k,n)$, defined as the increase in capacity share $\alpha_t(k,n)$ of tenant k to be applied in the following time step in cell n. This increase is defined in discrete steps of size Δ, so that the action can take three different values $a_t(k,n)\epsilon\{\Delta,0,\text{-}\Delta\}$.
- The reward $r_t(k)$ considers the weighted product of the SLA satisfaction factor of tenant k, the summation of the SLA satisfaction factors of the other tenants and the capacity utilization factor of tenant k, that is:

$$r_t(k)=\cdot\gamma_{SLA}(k)^{\varphi_1}\cdot\left(\frac{1}{K\text{-}1}\sum_{\substack{k'=1\\k'\neq k}}^{K}\gamma_{SLA}(k')\right)^{\varphi_2}\cdot\gamma_{ut}(k)^{\varphi_3},$$

Where γ_{SLA}(k) is the ratio between the provided and the requested capacity (representing the SLA satisfaction factor of tenant k, γ_{ut}(k) represents the capacity utilization factor and φ_x are the weights of each component.

Although the triggering of actions by each DQN agent is performed independently of others, the different DQN agents operate in a coordinated manner. It means that a collaborative reward is selected since the reward definition for tenant k considers the SLA satisfaction factor of both tenant k and the other tenants in the system in order to avoid selfish actions that would negatively affect the other tenants.

3.3.1.1.2 Radio resource provisioning in a multi-technology RAN

Motivation: Industrial scenarios, where only one operator usually prevails. In contrast to multi-tenancy scenarios, most devices of an industrial network take a fixed position in production tapes and a limited number of Automated Guided Vehicles (AGVs) move around the premises. The traffic of industrial applications is stringent in terms of latency, so that, services in industrial scenarios are more related with different types of ultra-reliable and low-latency communications.

On the other hand, *Radio resource provisioning in a multi-technology RAN* ML algorithm is an industrial scenario-centric problem that tries to solve the resource provisioning issue in this kind of private network. This network can be considered as a Standalone Non-Public Network (SNPN) managed by a unique private network operator that separates his offered services into several network slices. These network slices need to be provisioned with enough radio resources in order to preserve the service quality offered by the slices. However, radio resource provisioning becomes a complex task, especially in the considered scenario in which the radio access network is composed of non-3GPP access technologies (e.g., Wi-Fi) in addition to pure 3GPP ones (5G/LTE).

Mobile networks as such, and particularly the emergent 5G networks and all the use cases they bring forward, are rather complex systems to model. These novel systems comprising a great variety of network states and parameters to be configured are difficult to be optimized by traditional analytical modelling due to the lack of accurate mathematical models to address the problem, or because there is no sufficient domain knowledge, or professionals with vast experience are needed, driving up the network operational costs. Typically, the use of rule-based controllers alleviates this problem by enabling the description of expert knowledge in terms of basic rules that are applied to the network. In this regard, the controllers' configuration can be enhanced with the use of ML techniques. Furthermore, in order to make the radio resource provisioning problem more efficient, observations from the environment and the ability to react according to the environment changes are required. So that, controllers need to get adapted to the network dynamicity. Therefore, all the reasons above lead to use ML as the most suitable technique to address this problem.

Specifically, this algorithm tries to address the radio resource provisioning problem where network dynamics occur at a non-real time scale and require a periodical re-distribution of the available resources among the network slices. To carry out the network adaptation, the intended algorithm shall follow a closed loop automation strategy, comprising a continuous process that monitors, measures and evaluates network traffic, and then automatically acts to optimize the resource provisioning. From a design point of view, the closed loop automated slice provisioning can be seen as a controller that distributes the resources and traffic loads as a function of the current traffic demands and SLAs.

The controller's design (e.g. required inputs) depends on the performance requirements of the slices in the industrial environments. In addition, its configuration (e.g. rules) can be adjusted by an optimizer responsible for driving its operation according to the operator's high-level goals. Consequently, the algorithm will follow an optimizer/controller approach, where the optimization framework can be based on Deep Reinforcement Learning (DRL) in order to face the complexity of high-dimensional state and action spaces. Training in reinforcement learning involves an agent interacting with its environment. Particularly, the solution will be based in a multi-agent distributed approach, in which one agent will be deployed per cell and per slice. As well as the algorithm described above, due to the huge space of possible network configurations, the agents of this approach are also based on DQN algorithm, that combines Q-Learning with DNNs. DNNs are great non-linear function approximators, so that, they are used in DRL to approximate the Q-function. The connections among the layers of neurons that conform the neural network are configured with weights. As depicted in Figure 3-11, a multi-technology radio access network represents the environment. The agent state shall be defined with some performance indicators such as the slice throughput, the cell resource allocation, the slice resource quota and other indicators that are slice-specific (e.g. delay for URLLC slices). The action triggered by the agent will be the modification of the slice resource quota, which could be to increase, decrease or maintain the quota of allocated resources.

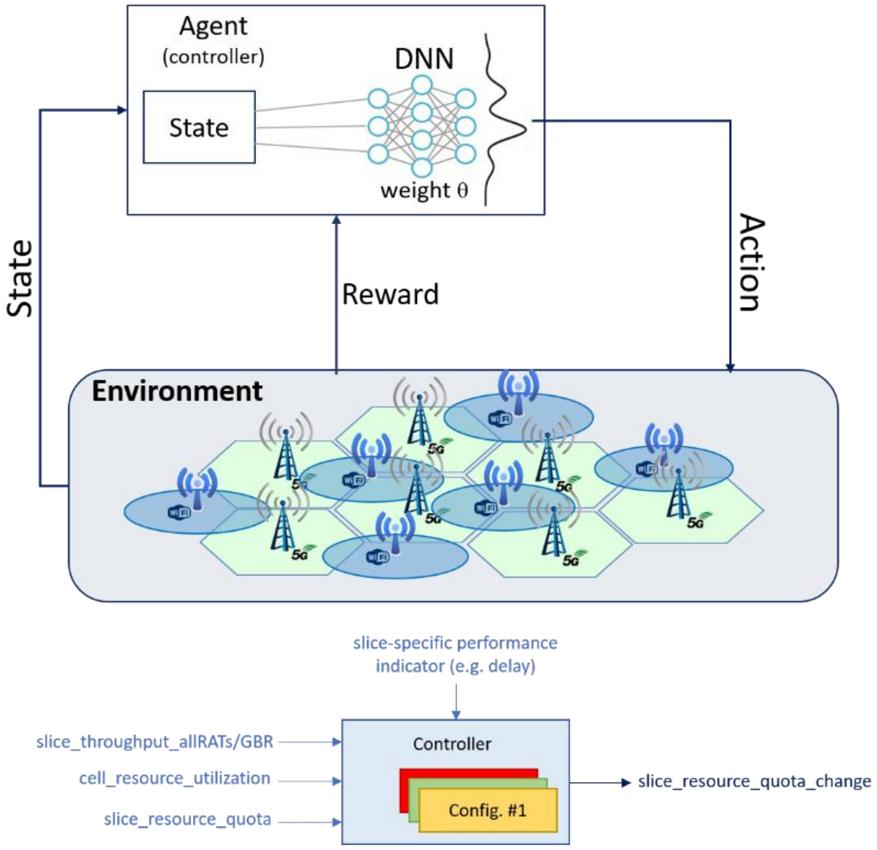

Agent
(controller)
DNN
State
weight θ
State
Reward
Action
Environment
slice-specific performance indicator (e.g. delay)
slice_throughput_allRATs/GBR
cell_resource_utilization
slice_resource_quota
Controller
Config. #1
slice_resource_quota_change

Figure 3-11 Radio resource provisioning using Reinforcement Learning

3.3.1.1.3 RL-empowered user association

Scrutinizing intelligent resource management methods to optimally control the available radio resources among the different network slices, is yet insufficient by itself to reassure the desired QoS that each network slice should provide to its serving vertical applications. In this respect, utilizing in an optimal manner the physical radio resources that pertain to a specific network slice, which can be either dedicated or shared with other network slices, can further ameliorate the provided QoS to the end-user devices. Given the heterogeneity of the underlying physical radio access/backhaul network with respect to the deployed base stations (e.g. picocells, femtocells, unmanned aerial vehicles), the mobility of the end-user devices, the uncertainties of the wireless links and the ever changing locally perceived interference, ML techniques can be used to adapt the respective physical resources in accordance with the network's dynamicity, targeting some QoS metric.

The aim of this section is to present a potential application of RL, capitalizing on the theory of learning automata, towards a resource conscious end-user device scheduling to the available base stations, and thus, the available frequency resources. Considering a heterogeneous and densely deployed radio access network, a real-time scheduling algorithm executed explicitly and in a distributed manner by the end-user devices that act as 'stochastic learning automata', is introduced in [181]. As illustrated in Figure 3-12, the devices can self-adapt and learn the most beneficial base station association, by observing the reward fed back from the communication's environment, i.e., the respectively communicating base station. At each iteration of the algorithm, the automata iteratively update the probability of communicating with a certain base station (i.e., action probability), based on the commonly used update rule referred to as linear reward-inaction $L_{R\text{-}I,}$ targeting to maximize their offered reward. The higher the cumulative action probability regarding a base station, the more frequently the specific association is chosen by the automaton at the successive iterations. The convergence of the algorithm is met whenever the total end-user devices' action probability for at least one base station reaches 1. Considering the modelling of the offered reward, this can, in turn, be properly designed to capture different QoS metrics, such as the communication delay, the interference induced to the remainder of the network, the achieved end-user device's data rate, as well as the congestion and the available frequency resources of each base station.

The learning approach introduced in [181], can be further extended to account for the capacity-limited wireless backhaul that interconnects the small cells (e.g. picocells, femtocells, unmanned aerial vehicles) with the core network, which greatly impacts the overall provided QoS. Indeed, several recent research works, i.e. [162], [25], advocate the importance of backhaul-aware resource management, though without investigating the potential benefit that ML can yield to the resource optimization procedure. In this context, novel end-to-end physical resource optimization mechanisms can be devised empowered by RL, enabling the self-adaptation of the end-user devices under the uncertainties of both the radio access and backhaul parts of the network.

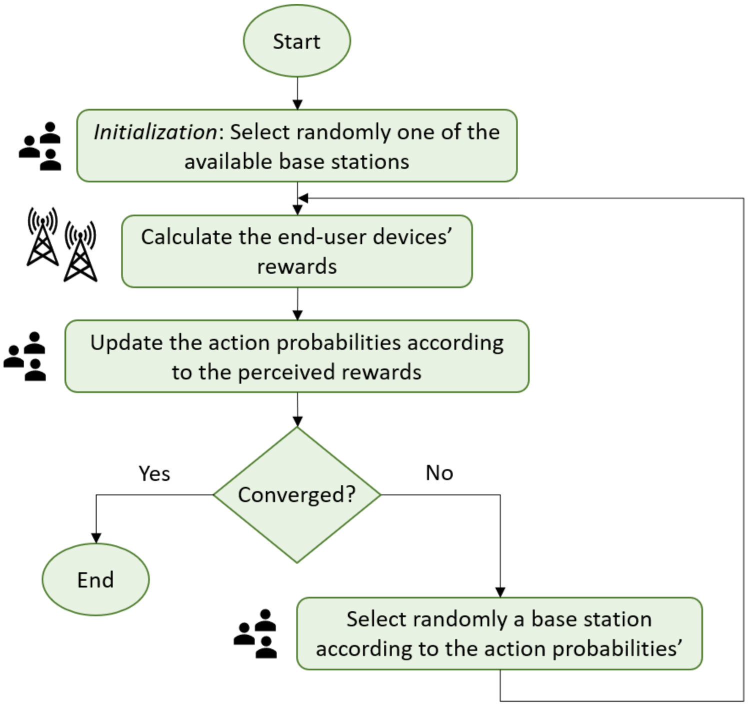

Start
Initialization: Select randomly one of the available base stations
Calculate the end-user devices' rewards
Update the action probabilities according to the perceived rewards
Yes
Converged?
No
End
Select randomly a base station according to the action probabilities'

Figure 3-12: Stochastic learning automata algorithm for distributed end-user device-to-base station scheduling.

3.3.1.2 Near real-time use cases (10 ms – 0.5 sec)

O-RAN defines the near real-time control loop as affecting control functions that operate in a time scale between 10 ms and 0.5 seconds. These functions are deployed in an O-RAN architecture as xApps connected to the near real-time RAN Intelligent Controller (RIC). Like O-RAN, 5GPPP is also working on developing AI/ML use cases that can work on these timescales.

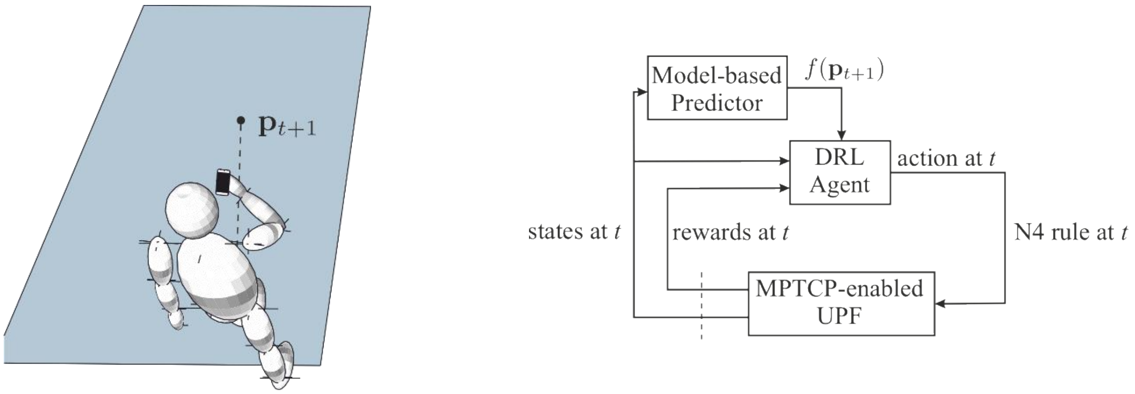
Table 3-6 describes two RAN near real-time related use cases presented in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-6 Use case for near-real time RAN aspects

Use Case	5GPPP Project	Additional references
Near-real-time traffic steering on eAT3S	5G-CLARITY	[327], [342]
Demand-Driven Power Allocation in 5G-enabled Wireless Networks	AFFORDABLE5G	[318]

3.3.1.2.1 Near-real-time traffic steering on eAT3S

The considered ML model for near-real-time (near-RT) traffic steering on deployed 3GPP (5G) and non-3GPP (Wi-Fi and LiFi) networks can be employed by utilizing the O-RAN reference architecture and interfaces [217]. For an ML-based near-RT traffic steering application, O-RAN E2 interface can be used to collect telemetry and push control actions. In the considered problem, the ML model firstly estimates UE mobility pattern as well as RSSI, link blockage events and predicted throughput. It can be considered as a model-based system that utilizes, 1) random mobility models, technology-specific channel models; and 2) telemetry data on UE connected cell ID and/or AP SSID, received signal strength (RSS) and downlink packet drop rate. Then, the ML model determines an optimal ATSSS rule (N4 rule) regarding the weights of an MPTCP scheduler that resides in user plane function (UPF) and allocates traffic onto available networks/paths. For the UE mobility estimation and traffic routing, a single deep reinforcement learning (DRL) agent is used along with a hybrid model-based and model-free system as shown in Figure 3-13. The model-free DRL system uses rewards and states from the provided ATSSS rule. The reward function can be the average goodput from all MPTCP sub-flows, and the states can represent round trip time (RTT), congestion window size, the number of active MPTCP sub-flows and the predicted metric from the model-based predictor.



(a) (b)

Figure 3-13 (a) An illustration of a new position vector, (b) a hybrid model-free and model-based DRL system

3.3.1.2.2 Demand-driven power allocation in 5G-enabled wireless networks

Given the large number of mobile users and/or densely-deployed network elements in 5G networks, overall interference defines an upper-bound in the network performance. In such complex environments, the optimal resource allocation inevitably becomes a non-convex problem, enforcing the idea of finding sub-optimal solutions [193], [159].

The main drawback of traditional methods for solving radio resource management (RRM) problems is the excessive computational cost, as well as the inability to generalize their solutions, thus becoming infeasible for large-scale cellular systems [159]. Coverage and Capacity Optimization (CCO) problem emerges when a considerable increase in network demands results into limited network capacity, thus degrading the available bandwidth per user. The increased network densification raises in turn significant challenges in the design and implementation of interference mitigation techniques, in order to ensure sufficient Quality of Service (QoS) to mobile users. Among a variety of CCO algorithms, power allocation has attracted scientific interest in the context of 5G design and deployment. Several power configuration algorithms have been proposed to optimize the network capacity, eliminate inter-cell interferences and regulate the coverage area of the network cells [159], [205], [308]. In this section, a Demand-Driven Power Allocation (DDPA) algorithm is formulated and implemented on realistic network configurations, aiming at the fulfillment of the user-specific requested throughput [23].

Network model: A network area accommodating M RUs is considered. Each RU has a total number of F resource blocks (RBs) that may be grouped in N channels. It is assumed that the F RBs are equally divided among the N channels, which in turn exhibit the same bandwidth B. Moreover, the m^{th} RU transmits over each channel n with a specified power level $P_{m,n}$, which is selected from a set of available L power levels. Finally, a maximum total power constraint is considered for each RU. Each user u located inside the network area may be associated with one channel n of a particular RU m. This user requests a service s from a set of available service classes S. Thus, a demand vector D_u is introduced to designate the requested service class of user u, expressed in terms of throughput.

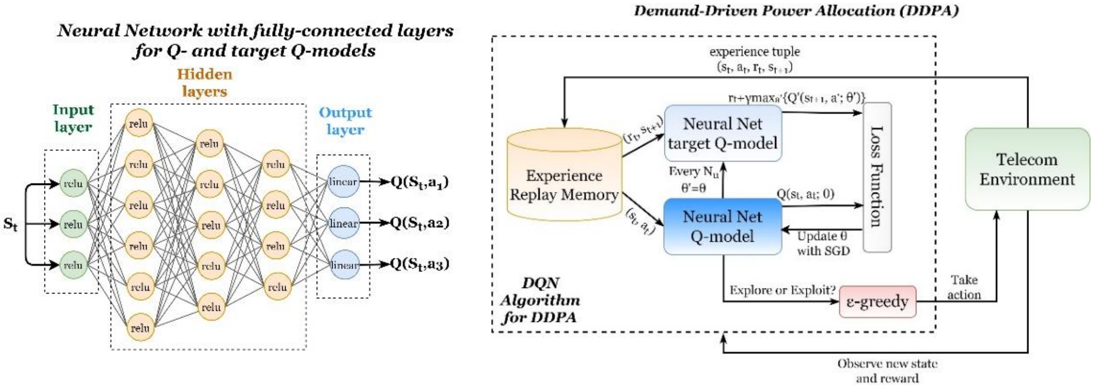


Figure 3-14 Neural network employed for training (left) and process of the DQL algorithm for DDPA (right).

DDPA problem with DRL: The target of the DDPA modelling methodology is to adjust the power vectors of the M RUs to maximize the throughput of each individual user according to the requested QoS class. In the presented DRL framework (see Figure 3-14), a central network entity observes the telecommunication environment and assigns power levels to all the channels of the RUs in the network area in order to minimize the difference between the allocated and the requested throughput. The DDPA training unfolds moving from an exclusively explorative to an explicitly exploitative phase (ε-greedy policy). The DRL descriptors are defined as:

State space: The environment state is efficiently described via three-fold information, namely the user-specific (i) Channel Quality Indicator (CQI) [99], (ii) the serving RU number and (iii) the allocated channel ID. For the association, the maximum throughput criterion is employed.

Action space: At a given time t, the agents assign a specific power level to the channels of each RU, i.e. $A_t = [(P_{1,1}, \dots, P_{1,N}), \dots, (P_{M,1}, \dots, P_{M,N})]$.

Reward system: Once the agent performs an action, a new network system state is triggered, leading to different CQIs and association configuration. The main goal of the rewarding function is to uniformly increase the allocated sum-rate among the individual users. Specifically, the agent receives either (i) a positive reward equal to the difference between the current and the previous sum-rate in case of sum-rate increase, (ii) a high-valued positive reward equal to the total requested throughput if the demands of all users are totally fulfilled or (iii) a zero value when the current action does not increase the sum-rate with respect to its previous value.

Sample results: The performance of the DDPA algorithm is evaluated by monitoring the training process using various hyper-parameters in the neural network configuration. Figure 3-15 depicts the accumulated reward for different number of hidden layers in the *Q*- and *target Q*-networks. In Figure 3-16, the performance of the DDPA algorithm is evaluated for several simulation scenarios (with varying demand vectors of users), including increasing user demands. The algorithm performance (ratio of fulfilled users) is verified by conducting Monte-Carlo simulations for different user positioning realizations within the network area. For comparison purposes, Figure 3-16 also depicts the performance metric resulted from two baseline methods, namely the Weighted Minimum Mean Square Error (WMMSE) algorithm and a fixed power allocation policy (Average Power), according to which each RU/channel transmits with the average/median power level as a reasonable trade-off between the achieved coverage and the potential interferences.

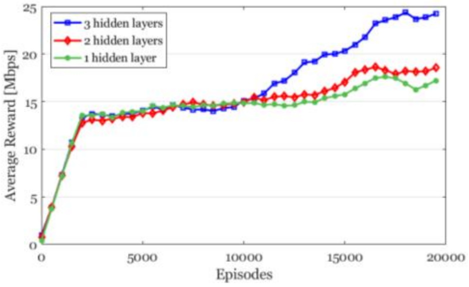


Figure 3-15 Learning curve for different number of hidden layers

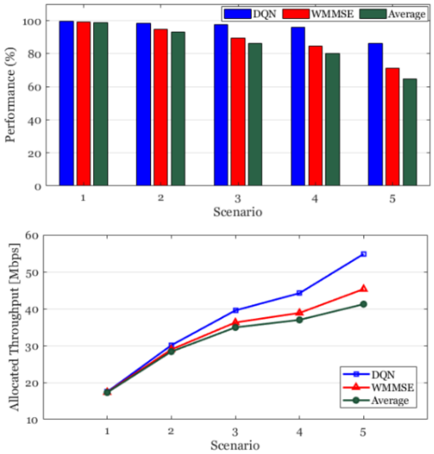


Figure 3-16 Comparison among methods: DDPA performance against WMMSE and Average methods for five different simulation scenarios (up) and total allocated throughput for the three methods (down).

3.3.1.3 Real-time uses cases (<10 ms)

Real-time use cases refer to applications of AI/ML techniques operating at control loops below 10 ms. Given the considered timescale the natural placement of these AI/ML techniques is the PHY or the MAC of the base station.

Table 3-7 includes three RAN real-time related use cases described in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-7 Use cases for real time RAN aspects

Use Case	5GPPP Project	Additional references
AI/ML for joint MAC scheduling across gNBs	LOCUS	[340]
AI/ML for Channel Modelling	ARIADNE	[242]
Prescriptive Analytics and Channel Estimation to optimize Reconfigurable Intelligent Surfaces (RIS)	ARIADNE	[242]

3.3.1.3.1 AI/ML for joint MAC scheduling across gNBs

There is a trend to reuse of spectrum over short distances as densification squeezes ever more cells into a given unit area and beamforming narrows the radial extent of each unit of coverage. The result is that interference becomes more challenging to manage. Failure to deal with inter-cell and inter-beam interference impacts capacity, shrinks cells and risks impairing the ability to meet challenging QoS targets.

There are various strategies to mitigate this. Interference coordination aims to minimize or avoid situations where a gNB schedules a transmission for a UE on PRBs which coincide with the PRBs that a nearby gNB schedules a transmission for another UE with overlapping beams such that the transmission for one of the UEs becomes a significant interference for the other UE. Collaborative transmission is another recourse where the gNBs work together to provide wanted transmissions rather than interference to the same UE and effectively become a distributed MIMO panel, although the aggressive timing coordination required for phases to be synchronized means that this is substantially harder to implement than interference coordination.

For every sub-frame, a gNB must decide on which beams and which PRBs it will schedule traffic for each UE. Different choices will have different implications for interference and thus QoS and capacity. But the resulting impact will depend on the corresponding choices of other nearby gNBs. How much interference will be experienced by each UE across a localized cluster of gNBs will be a result of the decisions made by each gNB of what PRBs each UE to be scheduled should be allocated. In an ideal world the gNBs would be equipped collectively to make an optimal decision. Being fully equipped would mean having perfect instantaneous knowledge of various quantities and states including what is awaiting scheduling for each UE, the latency tolerance of each of these quantities, the cost in terms of interference of scheduling any UE on one gNB to the same PRBs as another UE on a nearby gNB, and the cost in terms of lost service and revenue opportunity when scheduling is prevented. This is a lot to ask as it requires detailed knowledge of the channels between every gNB and every UE. But, more than this, these quantities and states are constantly changing and as the information ages it becomes less useful for making optimal decisions. There are mechanisms in wireless communications standards for exchange of information between nodes that supports this type of collaborative transmission. However, the time to communicate along with the need to constrain the size of data means that each node can only have incomplete information that is at least a little out of date. This is at odds with the need for each node to make immediate decisions on what to schedule in which PRBs and on what beam for every sub-frame.

AI and ML can help us here. For example, a model can learn to estimate the quantities and states upon which the ability to make optimal decisions depends. An ML model for the channels between each gNB and each UE would mean that the gNB could make more accurate decisions about the degree to which the UE in its current location and serving beam should be considered resilient to interference from a nearby gNB and its potentially overlapping beams and which UEs are conversely considered as located towards the “cell edge”. A specific mechanism to achieve this channel estimation is to rely on measurements of estimates of the user location, covered elsewhere in this paper. A model such as a convolutional neural network could be trained to learn the channel from each beam to each location. Additional features used as input to such a model might include the type of UE or its antenna system for example. This model could then be used to infer the channels from nearby gNBs and thus assign the UE accordingly.

This model-based channel estimation can be complemented by longer term agreements between the gNBs about which PRBs are reserved for each gNB to allocate to its cell edge UEs and which beams should be considered overlapping. AI can in turn enhance these longer-term decisions, specifically how many and which specific PRBs to reserve in each group and how the nature of such reservation varies depending on the degree to which neighbouring beams overlap so that the changing distribution of UEs and their communication needs are satisfied.

ML can also be employed to make probabilistic predictions about the communication needs and channel conditions of UEs in the near future, either directly or as a result of interim predictions such as location dynamics. If the channels or buffer states are estimated in advance, then more time is available for communication of the information and collective agreement between the gNBs of a more optimal allocation of UEs to PRBs, for example by allowing more putative PRB allocations to be considered. As the ML becomes more effective and able to predict further into the immediate future, so more optimal configurations can be found. Additionally, the expected traffic profile and its priority can be taken into account to allow subscriber SLA and operator KPI such as churn reduction and revenue to be targeted. One way to achieve this is to use autoencoding and generative models [50], [229], [265] to lower the dimensionality of the features that can influence buffer state.

More advanced AI methods can ultimately play a role here. The gNBs collaboratively choosing PRBs and beams for scheduling could be regarded as a cooperative game. Reinforcement learning approaches, such as based on multi-armed-bandit, are receiving much attention currently and have been shown to be able to yield promising results [309], [165]. The application to the cooperative game of collaborative scheduling is an interesting field for further research.

Thus, with application of AI and ML to the problem of collaborative scheduling, will come the ability to reliably meet the targets for more low latency services whilst also leaving larger volumes of capacity available for services with less stringent QoS needs.

3.3.1.3.2 AI/ML for channel modelling

With an increased number of antennas, wide bandwidth, mobility, uncertain channel models, high speed and special processing requirements, characterising new features of wireless channels for future communications becomes increasingly difficult and not suitable for traditional methods especially with much more potential AI applications [43], [143]. The idea of showing methods based on data does not substitute but adds to traditional methodologies based on a statistical model [32]. ML has been used to build path loss models to solve the issue of the complexity and time usage of the channels, as opposed to the conventional approach [36]. ML can also be utilized for channel features predictions, channel impulse response (CIR) modelling, multipath component clustering and channel parameter estimation [36]. In addition, ML can solve (but its applications are not limited to) the problems of large- and small-scale fading [37]. A useful ML algorithm for wireless communication that is aimed at the modelling and estimation of a channel is the SVM algorithm that can help predict path loss, make the process simpler and the results more accurate than the classical forecasting. Clustering is another key non-supervised training method used in the selection of multifaceted components, which have similar behaviours. It can enhance the processes of time variable channel model accuracy [37]. However, ML methodologies may take time to develop due to the complexity for aspects that influence channel modelling, including atmospheric effects and beam directionality [278].

The application of ML to improve the modelling and estimation of channels has its own challenges in over-fitting and under-fitting with a large amount of data, which in turn may lead to wrong estimates. The problems associated with blockage are most seen in ML techniques applied to the mmWave. This enables successful deep learning (DL) based predictive methods to become recognized widely because new communication features, such as complex scenarios with unknown channel models, high-speed and accurate processing can be well addressed [278], [266].

Overall, the DL has been reported to deliver a higher level of precision compared to decoding complexity and better efficiency. The channel estimation uses a DL algorithm, and with the aid of deep image processing techniques, super-resolution images and pilot position repair, unknown channel response values can be predicted with a minimum mean-square error (MMSE) [270]. Based on the well-known CNN of three-dimensional MIMO channels, prediction of the channel statistical characteristics can be defined. In addition the channel parameters used for the development of measuring data, including amplitude, delay, azimuth angle of departure (AAoD), elevation angle of departure (EAoD), azimuth angle of arrival (AoA), and elevation angle of arrival (EAoA) are generated by means of the wireless Insite program of radio tracing software that is used to training CNN [49]. Although, we discussed the useful DL approach for WLAN communication to model the channel and estimate the channel, it could take time to develop due to the difficulty of other channel modelling scenarios including fading, atmospheric effects, beam direction and a MIMO. The implementation of effective DL methods can therefore reduce complexity and increase accuracy than regular channel modelling.

3.3.1.3.3 Prescriptive analytics and channel estimation to optimize reconfigurable intelligent surfaces (RIS)

In this sub-section, we briefly share a novel application of prescriptive analytics, i.e. a short real-time optimization run that is devised on top of a predictive model of Reconfigurable Intelligent Surfaces (RIS), to assist in forming a beam in a spatial context. Here, an RIS refers to a reflector that has meta-surface unit(s) that may either stay passive or activated with an attached microcontroller. This prescriptive role of AI and ML in dynamic beamforming is being explored in [242]. Ongoing work seems to suggest that it may be possible to frame a predictive model of an RIS, that predicts the reflected angles or signal properties at receiver, where the input feature set includes an incident beam’s angle among others. Installing such a function at the transmitter AP, gives an AP the ability to discover possible beams to reach a UE whose location is assumed to be known, but with whom, a direct Line of Sight (LoS) is blocked. The proposed function may consist of constraints: 1) that bind or limit value ranges on input and output attributes – based on, e.g. what incident or reflected angles are physically possible in the particular context; and 2) imposing goals to minimize or maximize some attributes, including the predicted value (e.g. confidence for a class of interest, regression value or error measures). The prescriptive optimization run can thus be framed to enquire a prescription, i.e. the optimization prescribes input attribute values, which maximized the objective function composed of constraints. This specific application of prescriptive analytics is known as recipe generation.

With constraints injected in a short (milli-second range) optimization run, contextualized beams may be formed between the AP and the UE in Non-Line of Sight (NLoS) scenarios by making use of RIS-based predictions. This suggested scheme respects the limitations of current situation faced by the AP in terms of the feasibility of incident angles required to be projected on RIS to reach the UE with desired reflected angles and signal strength. This prescriptive use of AI and ML arguably also assists in deciding when to relinquish attempts for contextual beamforming using a particular RIS and quickly resort or delegate to other means of establishing connectivity.

Efficient yet accurate RIS channel estimation methods play a pivotal role in the joint design of active and passive beamforming, which in turns affects the achievable data rate during the sequel data transmission phase. Luckily, in most scenarios, especially at mmWave and THz bands, the wireless channels have the inherent property of being sparse, resulting from the poor scattering propagation environment. That is, the channel coefficient matrix is rank-deficient. Therefore, the channel estimation can be formulated as sparse signal recovery problems and be well addressed by off-the-grid compressive sensing algorithms. Relying on the estimates, either in the form of estimated channel parameters (i.e., angular parameters and path gains) or cascaded channel, RIS phase control matrix, base station beam former, UE combiner can be co-designed by resorting to convex optimization tools.

Moreover, the rough location information on the UE and environment objects, offered by existing localization systems, will further accelerate the channel estimation process and improve the RIS design and training beams used at the base station and UE. With the aid of compressive sending based channel estimation, higher resolution of parameter estimation is achieved, which in turn enhances the co-design of base station beam former, UE combiner, and the RIS phase control matrix.

However, challenges arise when the number of RIS elements goes extremely large and mutual coupling effect exists among the elements. The conventional model-based approach then faces difficulties in terms of the computational complexity, imprecise and even unknown channel models, etc. To address these issues, data-driven AI approaches have to be considered for directly mapping the received signals to the joint active and passive beam formers. The performance of such approaches will be affected by the available data volume, and their extendibility to new scenarios can be achieved by fine-tuning the parameters of the already trained AI models.

3.3.2 Transport networks, fronthaul and backhaul

The transport network domain has been subject of intense research during the 5G development phase, where the functional split adopted in the 5G RAN architecture directly impacted the capacities that need to be provided by the transport network. This work gave raise to differentiated transport network services such as the backhaul, the fronthaul or the midhaul. Optimizing the transport network is critical in 5G and beyond 5G networks, and thus AI/ML also finds a natural application in this network segment.

Table 3-8 describes five transport network related AI/ML use cases presented in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-8 Use cases for transport network related aspects

Use Case	5GPPP Project	Additional references
Triggering path computation based on AI/ML inputs	5GROWTH	[59]
ML-based Traffic management using programmable switches	5GROWTH	[60], [116]
Dynamic Load Balancing	TERAWAY	[328], [329]
Efficient per flow scheduling in programmable transport networks	5G-COMPLETE	[232], [233]
Determine optimal FH/BH functional split	5G-COMPLETE	[27], [30]

3.3.2.1 Triggering path computation based on AI/ML inputs

Traditional classification algorithms are used to protect services from possible failures and enhance the (transport) network stability using support vector machines [287], or to extract the common patterns from features such as message template sequence, frequency, seasonality and surge for predicting network switch failures using machine learning methods in [313]. One step further, more accurate results in using deep learning algorithms to detect network anomalies more specifically leveraging the feedforward neural network (FNN) and convolutional neural network (CNN) models [317], or to choose the best path combination for packet forwarding in switches using CNNs [188].

On the other hand, reinforcement learning algorithms can be exploited to detect radio link failures or degradations as in [150]. Additionally, such algorithms are adopted to choose the most stable path to increase resiliency to link failures in [212], to enhance the service restoration time via actor-critic-based service restoration (A2CSR) in [314], or to group network metrics into profiles required when handling network anomalies [267].

The decisions made by the AI/ML applications in detecting anomalies, forecasting traffic variations, etc., become the triggering event of subsequent actions tackled by the network operation functions (e.g., network orchestrator and controllers). Such actions aim at preserving the requirements (e.g., maximum tolerated delay, bandwidth, etc.) of the existing network connections as well as attaining the most efficient use of the resources. In such context, candidate actions conducted by the network management encompasses restoration, resource re-allocation and entirely/partly re-optimization. In all of them, a key functionality in the network management is the path computation. The path computation may operate as an on-line process (i.e., upon request) where, relying on an updated view of the network resources, it seeks for the network resources enabling to restore/re-allocate a connection (or bulk of connections) affected by a network failure or performance degradation (e.g., node and/or link down, increase of packet losses, etc.). Additionally, the path computation can also operate offline. That is, the path computation process is triggered, for instance, periodically or after several connections have been set up and released. The objective of such an offline path computation is to attain a more efficient use of the network resources that may entail recalculating and re-allocating the existing network connections. Regardless of the targeted action, the network operations handled at the orchestrator/controller needs to ensure that the connectivity service downtime is minimized. This is particularly relevant when re-allocating and re-optimizing network connections. In this regard, the network orchestrator/controller could consider strategies such as make-before-break (i.e., programming new computed resources prior to disconnect the old ones for a given connection).

3.3.2.2 ML-based traffic management using programmable switches

The use of ML-based frameworks is currently heavily investigated for optimizing the operation of 5G systems, in many cases focusing on traffic management at the data plane. For example, as different types of traffic flows have different levels of latency requirements, authors in [104] advocate for the use of an ML-based framework for semi-persistent elasticity- and latency-aware scheduling, i.e. an intelligent control plane that could prioritize, and schedule elastic traffic flows based on their requirements. In particular, ML can extract specific flow characteristics and the associated latency requirements and feed this information to schedulers. In another example [112], congestion prediction for elastic traffic, by means of Explicit Congestion Notification (ECN) feedback can be used to auto-tune parameters of active queue management mechanisms employed at the data plane.

At the same time, data plane programmability implies the switch capability to expose the packet processing logic to the control plane to support systematic, fast and comprehensive reconfiguration. An intelligent control plane, complemented by programmable data planes can promote zero-touch configuration and management and reprogramming of network operations, enabling self-optimization/management as envisioned in beyond 5G systems. In this direction ML-based network optimization over programmable data planes is being promoted by [238].

In particular, 5Growth adopts a P4-programmable transport data plane in the solution proposed in [115], enhancing programmability and isolation while it supports customization of selected queue management functions (e.g., Active Queue Management) [59]. Exploiting such capabilities and flexibility provided by data plane programmability, intelligent decision making can be applied at the control plane to enforce 1) performance isolation per network slice; and 2) application aware slice customization. To that end the project updated its orchestration and management plane to enable the enforcement of such QoS policies for all slices over the shared infrastructure [239]. Going one step further the project envisions a data-driven control plane, that autonomously can group slices and configure their priorities based on their requirements and the total link utilization and customize their scheduling and congestion control techniques on a per slice basis.

3.3.2.3 Dynamic load balancing

Optimizing the usage of network resources using ‘dynamic load balancing’ is far better than static round robin routing algorithm. However, dynamic load balancing is a mechanism that takes statistical parameters from each network device and evaluates the network traffic to modify the flow accordingly. The drawback in large networks is the performance cost, i.e. the overhead for collecting statistics from each device in the network and computing the best route. Moreover, dynamic load balancing needs high computing power to calculate the best path. Therefore, AI/ML is used to optimize the usage of resources given the complexity of the network and traffic growth.

The ‘mobile backhaul orchestrator’ is a prototype of control logic which make use of the SDN and ML technologies to:

- Measure and evaluate the existing network resources in each slice.
- Provide resources dynamically to new slices with relatively short lifespan with phases of setup, use and decommissioning.

- Resource reallocation from an existing slice to another existing slice or newly created slice without disturbing or with minimum disturbance to the operation of other slices.

Mobile backhaul orchestrator is used to create and manage network slices based on SDN and ML logic to utilize available resources on each slice efficiently. The design includes the modules depicted in the Figure 3-17. ML/AI engine will apply different algorithms to estimate optimal routes and guide the controller to decide which path to take to deliver packets with minimum delay. In this case we validate the network optimizer using ML techniques to evaluate and predict the congestion level of a link.

We chose ‘supervised learning’. The idea behind supervised learning is that, for some inputs, we want to have certain value as an output. The ML algorithms run based on the inputs until getting output values close enough to the target value. We use artificial neural network (ANN) to perform the tests. ANN is a network made of multiple neurons, where each neuron is a building block which takes one or more inputs and passes through some mathematical functions, in our case sigmoid function to produce an output. A sigmoid function is used to unbind the input from output. For implementing the neural network (NN), we use a powerful python library called NumPy. An NN has three layers: 1) the input layer; 2) the hidden layer; and 3) the output layer. We use three attributes, i.e. bandwidth, packet loss and hop count as input. The hidden layer is a layer between the input and output layers. There could be a single, or multiple hidden layers. For simplicity we use a single hidden layer (Figure 3-18).

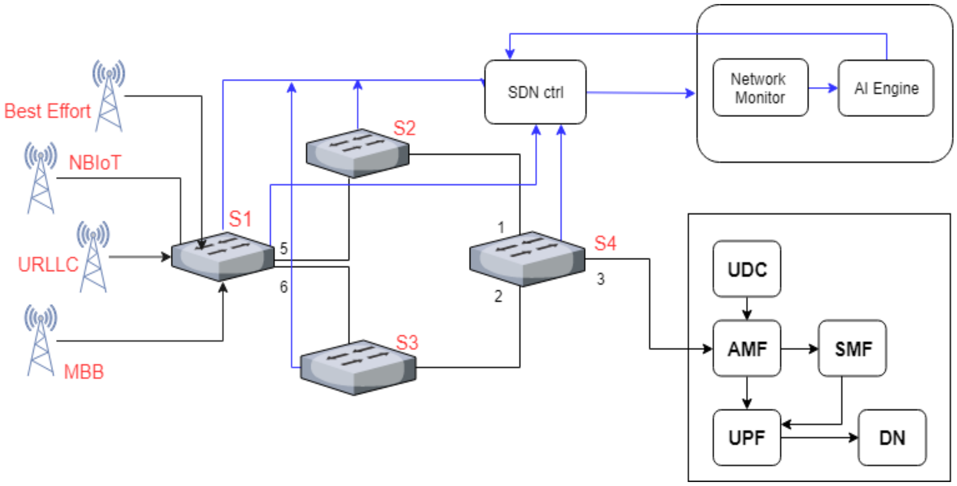


Figure 3-17 Mobile backhaul orchestration modules

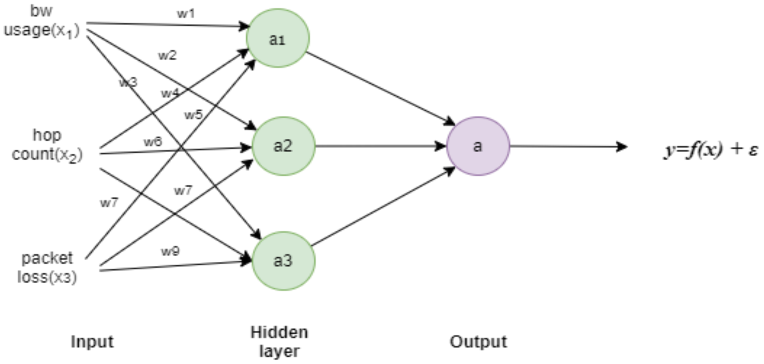


Figure 3-18: Single hidden layer NN

Initially each input is multiplied with some random number called weight (w) and are summed up and passed through an activation function to generate a neuron in the hidden layer. The generated hidden layer is now an input to the output layer or to another hidden layer and passes through an activation function and finally gets the output. The process of getting an output value from a given input is called the Feed Forward (FF) process. However, the output might not meet or close enough to the expected output, hence we perform the same process but backwards to find and replace the random weights with some reasonable values using partial derivatives and this process is called Backward Propagation (BP). Once we find new weights, we perform the same Feed Forward (FF) process to find the final output. In this case we might perform multiple FF and BP processes until we get the desired output before going to testing. The output value from the network features (inputs) is somewhere between 0 and 1. Value close enough to zero means the link in question is less congested and value close. The Network monitoring engine generates traffic statistics which describes the status of the ports of each switch.

About 1500 sample data was collected for one and half hours. Probably we could have done better data collection for better results but we needed better device. From the data collected 30% of it is used for training and the rest 60% of the data for testing.

3.3.2.4 AI-based flow management in 5G systems

5G networks are expected to operate in a highly heterogeneous environment supporting a diverse set of applications with different bandwidth, latency, mobility and reliability requirements [30]. To address these requirements and support the dynamic traffic patterns generated by different services, the transport nodes interfacing 5G RAN/CORE elements and other network technologies should be able to multiplex traffic flows from the RAN and optimize their resources. Optimization involves mapping the flows into separated traffic groups according to their transport KPI requirements when QoS is used for network slicing purposes.

Depending on the operator’s needs, allocation of network resources (e.g., bandwidth) to slices, can be performed either in a deterministic manner (i.e. allocation of a specific wavelength(s) or timeslot(s) that can be multiplexed in the frequency or time domain over a link) or statistical (i.e. logical multiplexing of VLAN connections over a physical channel). In this section, the concept of deterministic slicing is investigated where optical transport network resources (wavelengths, timeslots) are allocated to the appropriate slice leveraging the latest advances in optical transmission and switching technologies [200].

In this environment, a challenging problem that needs to be addressed is associated with the identification of the optimal routing, wavelength assignment and scheduling strategies. Traditionally, the wavelength resource assignment and scheduling problem is solved using a variety of techniques spanning from Integer linear programming [74] to heuristic algorithms [117], [100]. These algorithms assign to the end users’ appropriate wavelengths and timeslots to support their requirements.

Based on the observations above, we adopt a novel approach to solve the problem of allocating network resources to slices by combining both AI-based and traditional optimization techniques. While ML models scale well with the amount of data available, it has been shown that they are not as effective when applied to constraint modelling and planning. On the other hand, ILP schemes have been designed to handle constraints and preferences but do not scale well.

To solve this problem a hybrid scheme based on ML and ILPs is proposed. We initially estimate the total volume of demands that need to be scheduled within a specific time frame using an ML model. The output of this model is provided as input to a linear programming model that solves the relaxed Routing and Wavelength Assignment (RWA) sub-problem. Once solved, the optimal scheduling strategies are determined using a specially developed NN model. To achieve this, we first evaluate the performance of various scheduling heuristics under different network settings and traffic conditions. The performance of these algorithms is examined against a utility function capturing optical network resource efficiency and computational complexity. We have observed that under scarcity of optical network resources (i.e., high traffic loading scenarios) heuristics that allocate demands to the smallest available block (such as Best Fit) are more efficient. However, these gains come at the expense of increased computational cost. On the other hand, under low loading conditions less efficient schemes with very low complexity (such as Fit First) are preferable. Based on this analysis, the regions where each scheme provides the optimal performance can been determined.

In the second phase, a scheduler based on a Multilayer Perceptron (MLP)-based neural network is used. The MLP periodically decides on the most efficient resource allocation strategies (i.e., selection of the appropriate heuristic to be onboarded in the transport network (TN) edge node) taking into consideration the availability of resources, the type and characteristics of services and the loading conditions. Once the MLP has been executed it is able to identify the appropriate timeslot allocation strategy. The timeslot allocation policy scheme is continuously evaluated and if an alternative option has been recommended by the MLP, it is onboarded at the edge node.

The selection of the optimal scheduler to be onboarded at the edge node is treated as a classification task. Extensive simulations have been carried out to create the appropriate training dataset for the MLP NN. The architecture of the MLP-based scheduler is presented in Figure 3-19.

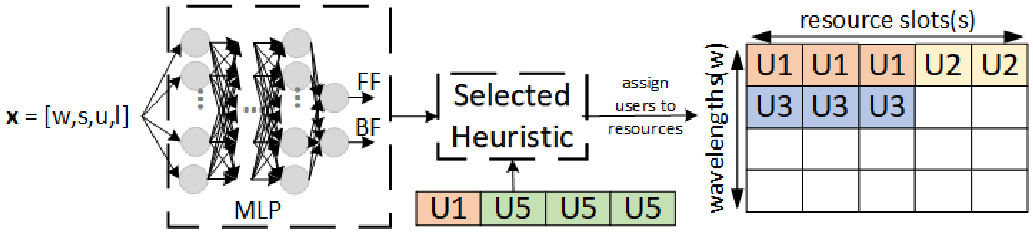


Figure 3-19 Flowchart of the MLP-based scheduler

In Figure 3-20, the MLP is compared against two conventional schedulers (i.e. the first fit and the best fit) in terms of weighted average utility. We note that by employing the MLP-based scheduler the average utility is increased for every combination of network parameter compared with the case when either the best fit or the first fit are employed.

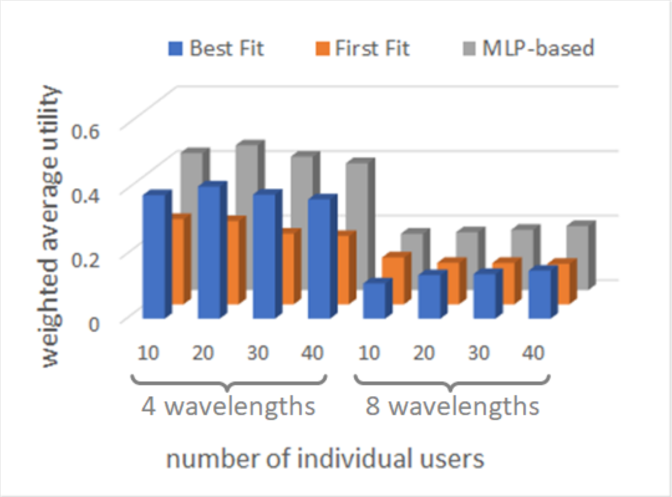


Figure 3-20 Weighted utility for the best fit, first fit and MLP-based scheduler for {10, 20, 30, 40} users and {4, 8} wavelengths

Once the MLP has been executed, it is able to identify the appropriate allocation strategy or scaling policy for the network. The resource allocation policy scheme is continuously evaluated and if an alternative option has been recommended by the MLP, it is on-boarded at the edge nodes.

Apart from edge transport nodes, the UPF has also a key role in the management of traffic flows as it acts as a termination point for several interfaces and protocol stacks including N3 (GTP-U) tunnels from the RAN, N9 for the interconnection of a chain of UPFs as well as N6 for interconnecting the system with an external data network. This introduces scalability issues due to the large number of packet detection rules required to support the relevant policies (i.e., end to end services) subject to limited network resources (e.g., memory in UPF-compliant network interfaces). To address this, [231] proposes the use of ML techniques aiming to compress the number of rules by taking into account the spatio-temporal correlation of mobile network traffic. The new compressed rules are then ported to the programmable optical transport edge nodes, which can be extended to process packets at line rate, adopting P4 implementations. To facilitate this, clustering schemes can be used taking advantage of the inherent correlation that exists between input traffic statistics, 5G topology and flow allocation decisions. From the implementation perspective this can be accomplished if the UPF can support complex forwarding and routing decisions. This can be achieved in practice by extending the optical transport with P4 capabilities that will allow dynamic allocation and configuration of resources as well as mixing of flows without affecting QoS. The P4 capable edge nodes are able to perform protocol adaptations (read the necessary filed in the GTP-U, decapsulate the payload, perform necessary classification, encapsulate the payload to the appropriate VXLAN and forward it to the appropriate port) at line rate.

With VxLAN, it is possible to aggregate an arbitrary number of flows by encapsulating them into a VxLAN tunnel, distinguished by a specific “virtual network indicator” (VNI). This way, we can create an overlay logical network that can be used to transfer transparently a set of flows with common characteristics or requirements. Through this approach the number of rules that are required to handle each flow separately can be reduced. A graphical representation of the overall process is shown in Figure 3-21 where a set of incoming flows can be classified using the developed machine learning model. Once classified, traffic management policies can be applied on the clustered flows to reduce the number of match action rules and, therefore, the size of the optimization problem.

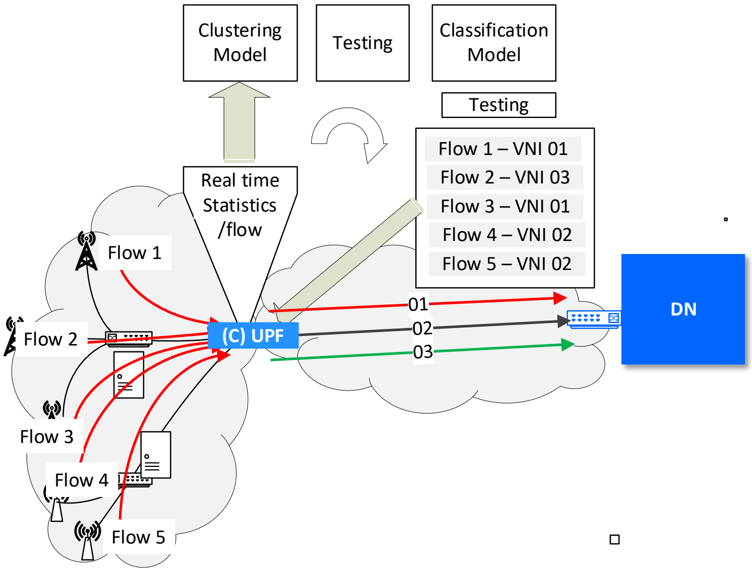
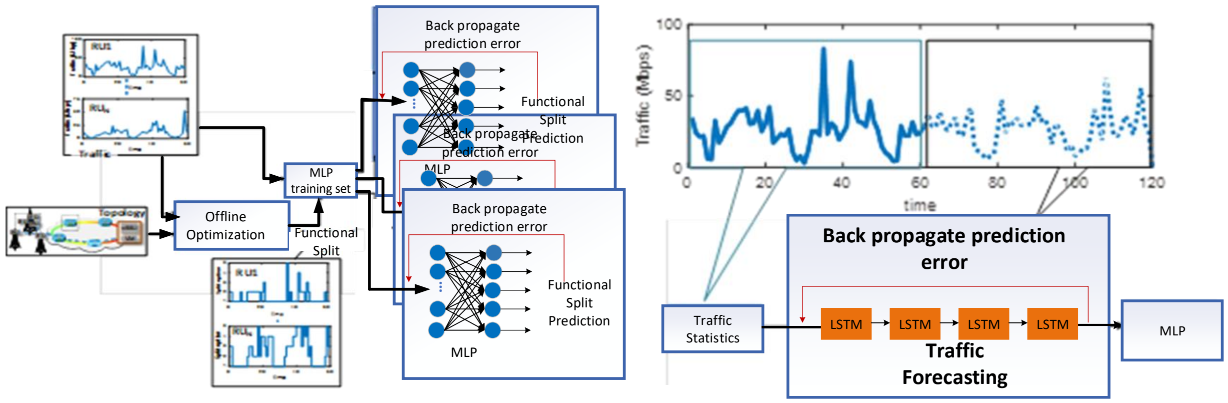


Figure 3-21 Monitoring, classification and tagging of network flows

3.3.2.5 Determining optimal functional split according to fronthaul/backhaul capacity

To successfully deploy 5G services there is a need for scalable control plane solutions able to manage and optimize the operation of a large number of highly heterogeneous network and compute elements, taking decisions related to: 1) *optimal embedding* of service requests and creation of service chains over the converged network resources [24], [12]; 2) *optimal infrastructure slicing* across heterogeneous network domains [31]; 3) *optimal sharing* of common resources in support of Information and Communication Technology (ICT) and vertical industry services [177]; and 4) *optimal fronthaul* deployment strategies including optimal placing of central units with respect to remote units, functional split selection etc. [101], [276]. These problems are traditionally solved by a centralized controller considering in many cases multiple objectives and constraints (ranging from Capital and Operational Expenditure minimization, energy consumption, latency, resource availability, etc.), adopting a variety of mathematical modelling frameworks based on integer linear [183] and non-linear [30] programming (ILP, NLP), stochastic linear and nonlinear programming formulations [178]. Although these schemes can be effectively used to identify the optimal operational points of the whole system, their increased computational complexity and slow convergence time makes them unsuitable for real time network deployments.

To cope with the increasing computational complexity inherent in these models, less computationally intensive online tools based on neural networks (NN) can be adopted. More specifically, offline optimization models coupled with history measurements from operational networks can be used to create a set containing the optimal design policies for converged 5G network environments [178]. Multilayer Perceptron (MLP)-based NN can then use the output of the ILP as a training set. Once NNs have been trained, they can be used by the centralized controller for real time optimal decision making. A typical example includes allocation of resources to support fronthaul and backhaul services over a converged high capacity and flexibility optical transport network environment. To solve the problem of optimizing fronthaul services, the NN model can be used to identify, in real time, the optimal functional split for each RU, the MEC facilities where the baseband unit can be hosted as well as the transport network resources for the interconnection of the remote antennas with the MEC server.



a) b)

Figure 3-22 a) Construction of the training set b) NN model-based LSTM and MLP for the optimization of the 5G network in the upcoming time instants.

A high-level view of this process is shown in Figure 3-22 a) for a specific case where the MLP NN is used to identify the optimal split per RU. To achieve this, a training set combining data from history traffic statistics as well as data extracted from the offline – Integer Linear Programming (ILP) based - optimization framework is considered.

The identification of optimal MLP-NN architecture is also a critical point to be addressed. To tackle this challenge an algorithmic approach has been developed that can identify an MLP network that maps any input x to the corresponding output y. Output y is obtained from the solution of the corresponding ILP formulation, while x represents the set of history observations. As an example, consider the scenario for which we apply to the MLP a training set that comprises a set of pairs (x, y), where x represents the traffic statistics for a particular RU at a given point in time, while y represents the functional split. The optimal functional split per RU over time can be obtained through the solution of the ILP model. This training set is given as input to the MLP neural network in order to learn how to map each input x to the corresponding output y. Once the system is trained, MLP can predict the functional split given any new data without solving the corresponding ILP. The parameters of the MLP model (batch size, number of hidden layers etc) can be derived applying simple heuristics that can maximize the prediction accuracy. Once the model is trained, the MLP-NN model is combined with a LSTM NN model used for traffic forecasting. This aims at identifying the optimal operating conditions for the 5G infrastructure in the upcoming time periods.

The performance of the proposed NN scheme is compared to the ILP based optimization approach in terms of total network power consumption. It is observed in Figure 3-23 that the power consumption over time for both schemes takes very close values, indicating the effectiveness of the proposed NN scheme to identify the optimal operational strategies of every network element. This clearly shows that online optimal service provisioning can be achieved taking a low complexity approach adopting ML techniques that can be trained to take very close to optimal decisions in real time. In this context, the training process plays a key role and can be performed taking advantage of the optimal decisions provided through offline tools based on ILP.

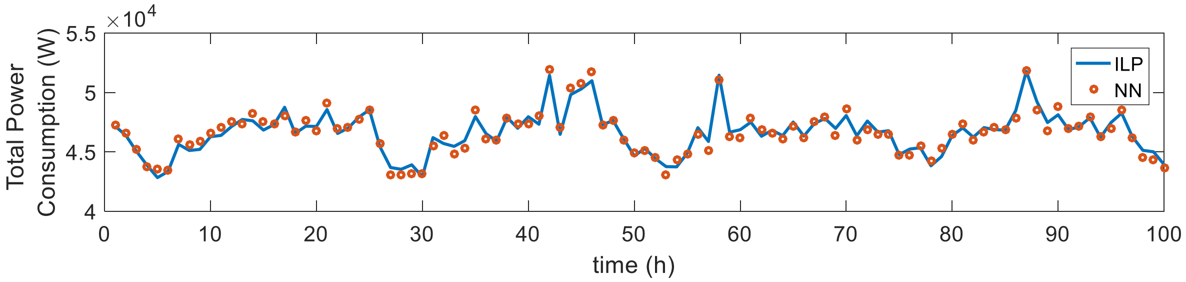


Figure 3-23: Total Power consumption when applying the ILP and the proposed NN scheme

3.3.3 NFV infrastructures

A key design principle for 5G, which will continue to guide the design of future generations, has been the adoption of softwarization techniques that have proven extremely efficient in the IT domain. Network Function Virtualization (NFV) is the embodiment of this design principle, which has resulted in the definition of a management framework that Mobile Network Operators (MNOs) can use to deploy network services as a set of concatenated virtualized network functions, instantiated over a general-purpose compute infrastructure. Optimizing the compute NFV infrastructure (NFVI) it is therefore key to the operation of future networks, and a natural application domain for AI/ML techniques that can be used to decide how common hardware resources can be allocated across a varying set of virtualised network functions.

Table 3-9 summarizes three NFV related AI/ML use cases discussed in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-9 Use cases for virtual network infrastructure related aspects

Use Case	5GPPP Project	Additional references
Federated Learning across MEC and NFV orchestrators	5GZORRO	[333], [334]
Resource allocation for Service Function Chaining	5G-CLARITY	[325]
Dynamic Resource Allocation in NFV infrastructures	5G-VINNI	[136]

3.3.3.1 Federated learning across MEC and NFV orchestrators

In this section we describe the techniques for automated resource and service management using Federated Learning (FL) models (Section 2.4.4). In the edge computing research direction, the network slices are extended by edge resources and services, such as the MEC platform, the MEC applications, the User Plane Function (UPF), the Radio Access Network (RAN) or even Cloud-native compute, network or storage functions. As these resources lie closer to the user than the NFV MANO, a scheme that is currently followed is the presence of a dedicated management entity on the edge for the resource lifecycle management, which includes instantiation, decommissioning and other functionalities. Such entity is called Edge Orchestration Platform (EOP) and provides distributed processing and storage capabilities that reduce the network management complexity. EOP is also able to interact with the NFV MANO, as depicted in Figure 3-24.

The interaction is enabled by FL models that are trained and executed on the edge level. Furthermore, the objective of using the FL models is to improve the efficiency and provide a high-level of network automation. The reasoning behind the choice of FL models as a ML technique lies in the presence of multiple edge Points-of-Presence (PoP) in different distributed locations that reside closer to the user than the NFV MANO. Concretely, the FL models provide 1) data processing and caching in each edge PoP as well as 2) automation in the formation and management of network slices. In this latter case, the EOP receives configuration instructions from the NFV MANO for slice instantiation or extension on the edge level.

In terms of employed ML techniques, the FL models are initially based on offline learning using unsupervised techniques, as data clustering, to identify the types of edge resources in each edge PoP. Upon the completion of offline training, additional supervised training techniques are used to translate high-level intents from NFV MANO into concrete instructions on how to instantiate and connect edge resources in each edge PoP. This is based on Natural Language Processing (NLP) that is performed by the EOP. Overall, the procedure that is followed is divided into three individual steps.

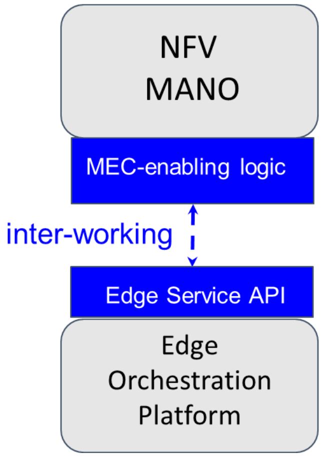


Figure 3-24 - Automated interaction between Edge and NFV MANO through FL models

Initially intent-based policies are specified, which provide high-level abstractions of edge resources and services by service consumers. Intents aid in hiding complexity, technology- and vendor-specific details. As a second step, appropriate APIs on the edge are used to receive these intents and then 1) translate abstractions of resources and services coming from the NFV MANO into edge instructions or configurations as well as 2) respond to requests and provide relevant information. To this end, the ETSI MEC ISG has provided an initial set of API's[3] to facilitate this interaction. The APIs are registered and discovered over the Mp1 reference point defined in ETSI MEC architecture [96].

The third step concerns the training of the FL models. In this scenario, each EOP performs first the offline FL model training using local data about the different resource and service types responding to the intents that originate from the NFV MANO. Then, during the offline training phase, data clustering techniques are employed to categorize edge resources based on specific features, such as their aforementioned type (i.e. MEC platform, MEC applications, UPF, RAN) as well as generic features as their location (i.e. longitude, latitude). Finally NLP techniques are used by the EOP that is associated to each edge PoP, in order to translate the intents from the NFV MANO into the involved resource clusters. Then, the associated resources are instantiated and connected using virtual links based on the intent translation.

A next step to provide further interoperability concerns the inclusion of these functionalities into a dedicated micro-service for each EOP. Such service would allow an even more efficient EOP response to resource/service discovery queries about each edge PoP service-layer and/or resource-layer status (available services/resources, active and historical service bindings).

3.3.3.2 Resource allocation for Service Function Chaining

One of the main tasks needed by the Service Function Chain (SFC) use is how the orchestrator can efficiently allocate network resources, the decision on the amount of computing resources to assign a VNF. The other is the placement of this VNF.

The resource allocation and placement of VNFs are two problems addressed widely in the related literature [26], [133]. The application of ML to solve these tasks is proved to be feasible in some scenarios. Even though it may provide suboptimal solutions, authors in [285] identify the use of ML to provide initial solutions for heuristics algorithms.

[3] https://forge.etsi.org/rep/mec

However, these ML-based proposals focus on individual tasks. There are few studies on coordinating the different management tasks by using ML [262], [261], which in most cases concentrate on guaranteeing the delay budget of the SFCs. Furthermore, there exist applications and services which rely on other performance indicators. Thus, further research effort is needed to design ML strategies which take into account additional QoS parameters such as energy consumption, reliability or jitter.

Related to the application of ML to aforementioned problems, authors in [262] identify the following challenges to apply ML for solving resource allocation problems in NFV:

- To increase the accuracy of the solutions provided by ML, since they may not be optimal.
- To reduce the training time of the ML models.
- To apply new ML techniques which don’t rely on Deep Reinforcement Learning.

In addition to the VNF resource allocation and scaling tasks, the mobility of users may imply for certain services collocated at the edge of the network to migrate towards another MEC infrastructure. This may involve the reallocation of computation and network resources, and the transfer of one or various VNFs’ states. In order to avoid the service interruption or degrading the QoE experienced by the user, the network could use ML to predict the trajectory of the user or even the future resource demand [160]. This way, the network could identify the most suitable MEC which guarantees the service level required. Moreover, from previous migration episodes, the network could calculate when the migration process should start, so both the resource reallocation and state transfer durations don’t impact on the user.

3.3.3.3 Dynamic resource allocation in NFV infrastructures

With network virtualization, network functions can be deployed over commercial off-the-shelf computers such as industrial data centres or in cloud as a service infrastructure. Furthermore, network slicing promises to provision resources to diversified services with distinct requirements over one heterogeneous infrastructure. These new technologies motivate for a solution to the management and orchestration of the underlying physical network resources. NFV MANO focuses on the specific management and orchestration of virtualization tasks, necessary to provide a network service and to meet the service level agreement. NFV MANO is in charge of providing the functionality required for the provisioning of VNFs, and the related operations, such as the configuration of the VNFs and the infrastructure these VNFs run on. It includes the orchestration and management of physical and software resources that support the virtualized infrastructure, and the life-cycle management of VNFs.

Traditionally, one of the most popular ways to address resource provisioning is threshold-based reactive approaches, where resources are added or removed if the network's conditions reach a certain predefined-thresholds. Although this idea provides a simple and scalable solution to dynamic resource allocation, threshold-based criteria tend to over-provision and under-utilize network equipment, leading to high costs for the infrastructure provider or the infrastructure tenant, and make the management of dynamic traffic and deployment of new types of services difficult as network traffic models must be elaborated beforehand. This may limit the reaction to current deployment on adapting to new situations, not seen before by the hand-crafted models. This approach is the default solution implemented in most of current networks and on NFV software tools like open-source MANO (OSM).

To overcome these limitations, ML approaches can be leveraged to provide either assistance to the traditional threshold-based solutions or to directly achieve a zero-touch network, i.e., a network that autonomously provide/remove resources to its VNFs and scales the services up and down accordingly. ML solutions that fall into the first class are generally based on event forecasting, and employ techniques like time-series analysis, data classification or data clustering. These techniques are used to assess the current network situation and understand where the network is heading to. An example of this type of solutions can be found in [256], where decision trees are used to classify cloud centre resource demand, and a one-to-one hardcoded mapping between the different categories and the resources needed is used to automatically scale instances. DNN classification is used in [259] to classify the network traffic and predict whether the different types of traffic are going to increase/decrease in the future. Each class is then mapped to a concrete number of VNFs that must be instantiated to be able to cope with the expected traffic load. Authors use historical labelled traffic data to train the proposed algorithm, which makes this solution less resilient to new traffic features. Based in these solutions, we see how these approaches are reactive in nature, i.e., prior assumptions (traffic mapping to resources) have to be made and the network reacts unequivocally based on these assumptions.

Under the second category fall the ML solutions that learn to autonomously scale up-down and instantiate new VNFs instances (vertical and horizontal scaling). Given the complex nature of self-governing networks, most of the works that fall in this category use deep reinforcement learning to obtain a reliable solution. For example, in [292], a DRL solution based on DDQN is presented for multi-tenant cross-slice resource orchestration, where a discrete number of communication and computation resources have to be allocated to different slice tenants. Another example is [136], where a novel DRL algorithm is presented for MANO. In this work, the authors study how an agent placed at the central unit can learn to scale vertically (add processing power and storage), horizontally (instantiate new VNFs), or to offload (send the VNFs to other cloud) based on the system state. The proposed solution show great improvement over state-of-the-art approaches. This type of solution is proactive as they do not consider any heuristic in order to take actions.

3.3.4 End-to-end slicing

Network slicing allows network infrastructure to be divided into different logical networks devoted to customized services and applications. In essence, network slicing is a key enabler of future cellular networks. By running fully or partially isolated logical networks on the same physical infrastructure, a substantial resource multiplexing gain can be attained. It is clear thus, that smart techniques are required to optimize the underlying RAN, transport and compute infrastructure. The application of AI/ML techniques, with their innate ability to predict and forecast events or demand, to build management functions for end-to-end network slices is a very active field of research.

Table 3-10 describes six network slicing related AI/ML use cases discussed in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-10 Use cases for E2E network slicing related aspects

Use Case	5GPPP Project	Additional references
Automated end-to-end service assurance	5GVINNI	[320], [321]
Proactive resource reservation in e2e slicing	5GSOLUTIONS	[322]
Joint slice admission control and resource allocation	MONB5G	[102], [103]
Joint slice-based demand prediction and resource allocation	5G-TOURS	[76], [77]
AI assisted slice isolation	5GZORRO	[330], [335]
AI/ML-based Decision Making for Slice Optimisation	5GENESIS	[235], [236]

3.3.4.1 Automated end-to-end service assurance

Service assurance is a key element to be autonomized in 5G. Due to the high complexity and diversity of the network elements and services to be managed and operated, service assurance automation becomes critical and mandatory rather than optional or preferred in 5G networks. Closed-loop control is one pillar of the autonomous management that has been applied and successfully verified in many industries over a century. It uses feedback to monitor and regulate itself to achieve a specific target. Applying closed-loop control to telecommunications was mainly focused on individual network elements or protocols (e.g., self-healing in BGP, self-configuration in TCP congestion control, or self-optimization in radio resource management). It is challenging to apply closed-loop control to autonomize end-to-end service management in mobile networks. Standardization bodies like ETSI ZSM have proposed a service-based zero-touch network service management and operation architecture and consider closed-loop as a powerful candidate for zero-touch automation.

Most closed-loop models consist of four stages: Monitor, Analyse, Plan, and Execute. ML and AI, as one key enabler for closed-loop control, play an important role in stages like monitoring, analysing, and planning. Various AI/ML models and algorithms are developed to solve service-assurance problems at different closed-loop stages, especially in analysing and planning, such as conventional anomaly detection, root cause analysis, and prediction. In addition, a close relationship exists between monitoring and analysing, whose integration is highly beneficial. In [195], different levels of monitoring-analysing cooperation were reviewed. AI/ML mechanisms like 'feature selection' via principal component analysis (PCA) [290], decision tree, random forest and deep learning help to decide in the monitoring policy what features are monitored and thus to reduce the monitoring cost. Similarly, 'flow selection' empowered by network tomography, maximum likelihood estimation and Bayesian inference optimizes the number of monitored traffic flows that can infer the entire network in the E2E service monitoring. Network-tomography-based inference methods, as described in section 2.4.2, are also valuable to optimize probe deployments as it is unrealistic to deploy monitoring probes everywhere in the network. In addition, correlation analysis is important in the NFV and network slicing environment with a high level of resource sharing, which not only complicates the data analytics but also introduces complex correlation relationships into the collected data sets. Correlation analysis is beneficial for improving service assurance performance or reducing the monitoring cost. However, correlation analysis is under-developed and needs more attention.

Within the context of 3GPP a communication service can be composed of multiple slices (3GPP TS28.530 [6]) and multiple communication service instances (3GPP TR 28.800 [1]). The former can be defined as a multi-slice communication service whereas the latter can be defined as a multi-service communication service. In general, the SLA of a multi-slice communication service (as well as multi-service communication service) describes the overall service level objectives of the composed slices or services. Each network slice instance (as well as communication service instance) will have its specific Quality of Service (QoS) and Quality of Experience (QoE) requirements. If the same Network Slice (instance) is shared across multiple communication services, which is also a feature defined by 3GPP, the corresponding management data and SLA objectives of the shared Network Slice for each communication service needs to be fetched correctly and assured. To manage the end-to-end Network Slicing, 3GPP defines the management functions (CSMF, NSMF, NSSMF) which, however, do not fully support the end-to-end assurance of multi-slice, multi-service communication service management.

To assure the end-to-end multi-slice communication service (as well as multi-service communication service), management data and analytics data play a critical role. In general, a cross-slice and cross-service communication service management requires the management data (e.g., performance management (PM) and fault management (FM) defined in 3GPP) for QoS management and Analytics data (e.g., NWDAF 3GPP TS 23.288 [4] and MDAS 3GPP TS 28.533 [7]) for QoE management of each composed slice or service of a communication service. If the end-to-end communication service (i.e., any composed Network Slice instance or communication service) does not meet the QoS/QoE requirements during the E2E SLA assurance of the service, a modification of the service on one or more Network Slice or communication instances should be triggered. AI/ML techniques can be used for this purpose. A Network Slice instance modification includes a network slice subnet instance and a network function modification. The modification or reconfiguration may include the scaling in/out of the resource utilization of a Network Slice Instance, or the migration of a composed network slice instance.

3.3.4.2 Proactive resource reservation in E2E slicing

Nowadays, resource assignment to VNFs is a reactive process based on hysteresis thresholding policies. However, such policies inefficiently over-provision network resources and might under-utilize available resources. This hence necessitates the need for proactive, data-driven solutions that can provide cost-efficient network resource utilization. This would be likely achieved by anticipating future capacity needs and timely allocating resources based on the time-varying demands. In this regard, we here envisage a data analytics-driven approach that can leverage tools from AI and machine learning for anticipatory allocation of resources in cognitive mobile networks. This envisioned approach is indeed driven by the recent advance of machine learning solutions, and the availability of big data for training. The ultimate goal is leveraging historical networks and users' information to provide operators and service providers (SPs) with adequate knowledge about the capacity needed to accommodate future demands per network slice and proactively allocate resources in response to this anticipatory information.

We now discuss the key pillars and aspects to be factored in towards AI-driven slice/VNF resource management and allocation. First, we refer to the main elements that need to be forecasted in order to enable proactive resource allocation. For example, in addition to traffic forecasting, it is of utmost importance to account for the cloud price variation. It is worth noting that the target of SPs is to purchase cloud resources to provide NFV services to customers while minimizing the expense of NFV providers. Moreover, in practice, different types of resource reservation might undergo different charging policies. On-demand reservation, for instance, can be performed at a fixed cost in accordance with abrupt changes to the traffic demand, while spot virtual machine (VM) prices are susceptible to price variations with time. Such cloud price variations might affect the overall process of capacity estimation and cost minimization, and thus should be factored in and forecasted. As detailed in Section 2.1.3, traffic demands as well as cloud price variation can be efficiently predicted using tools from ML such as recurrent neural networks (RNNs), particularly, using long short term memory (LSTM) [299].

Second, the architecture of the cellular networks and location of datacentres hosting VNFs would also play a vital role in the resource assignment and reservation. For instance, overprovisioning resources in core datacentres is relatively cheap, while service level agreement (SLA) violation might affect a large user population, and hence, this is more expensive at the network core. In contrast, violations of SLAs at edge datacentres will impact a limited number of subscribers in smaller areas and are thus less costly. Meanwhile, deploying resources in the proximity of the radio access at the network edge is typically expensive. Therefore, accounting for this inherent trade-off between SLA violations and cost of overprovisioning and relating it to the network level, i.e., core versus edge, are of paramount importance when estimating required capacity.

Finally, we comment on two potential approaches that could be adopted to address the proactive resource reservation problem, specifically, one stage and two stage approaches. For the former, a learning algorithm can be adopted to directly return a forecast of the capacity required to accommodate the future demands for services associated to specific network slices. For instance, a data analytics tool named DeepCog was proposed in [77] for the cognitive management of resources in 5G systems. DeepCog could forecast the capacity needed to accommodate future traffic demands within individual network slices while accounting for the operator's desired balance between resource overprovisioning and service request violations (i.e., allocating less resources than required). Based on real-world measurements, the use of deep learning techniques for resource orchestration such as DeepCog allowed for substantial reduction of operating expenses with respect to resource allocation solutions based on legacy mobile traffic predictors. In contrast, for the two-stage approach, the traffic forecasting is first performed then followed by the capacity estimation process. As an example, the authors in [68] addressed the issue of resource provisioning for end-to-end dynamic slicing based on datasets stemming from a live cellular network endowed with traffic probes. In their approach, they first introduced slices' traffics predictor based on a gated recurrent unit (GRU). They then built joint multi-slice deep neural networks and trained them to estimate the required resources based on the traffic per slice, while not violating the rate-based SLA and resource bounds-based SLA. Such different one and two stage approaches might achieve different, possibly comparable, levels of performance and undergo different complexity of implementations.

3.3.4.3 Continuous multi-objective reinforcement learning for joint slice admission control and resource allocation

To enable zero-touch end-to-end network slicing, a closed-loop modular architecture can be designed, wherein a deep reinforcement learning (DRL) agent over all technological domains (e.g., RAN, cloud, edge, core) performs end-to-end slice joint admission control and resource allocation/reconfiguration in either pseudo real-time or non-real-time granularities. As shown in Figure 3-25, the DRL controller is fed by key performance indicators (KPIs) measurement data that could stem either from a simulated environment or a live network.

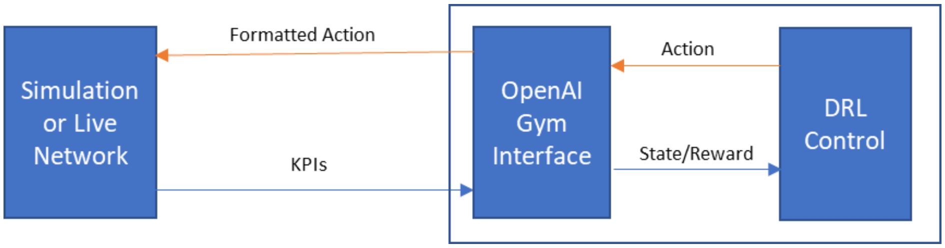


Figure 3-25 Slice-level admission control and resource allocation loop per domain

Moreover, given that both the measured KPIs and the requested actions are usually continuous (i.e., with non-discrete values), the adoption of continuous deep reinforcement learning (DRL) algorithms is needed. This family of control schemes encompasses several variants, such as Deep Deterministic Policy Gradient (DDPG) and Twin Delayed DDPG (TD3). Each of which presents a different architecture in terms of actor and critic deep neural networks (DNNs), and as shown in Figure 3-26, are combining both Q-value and policy-based learnings since there is an infinite number of actions and/or states to estimate the values for in the continuous case and hence value-based approaches are way too expensive computationally, which is referred to as the *curse of dimensionality*.

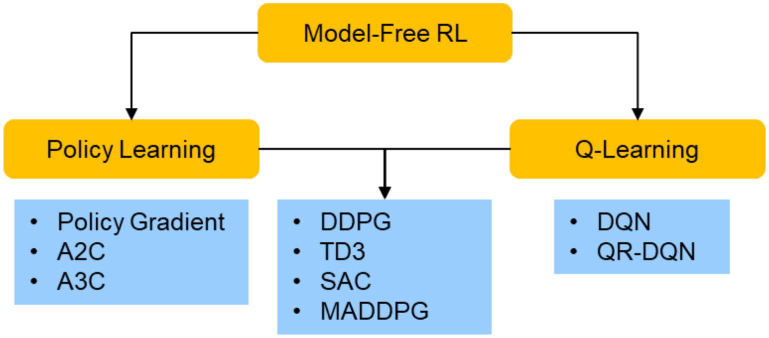


Figure 3-26 Model-free reinforcement learning

Environment

The OpenAI Gym-based environment interacts with the simulated or live networks to either get the necessary measurement data to form the state space, or to enforce the action into them. Typically, the environment is including the following components:

- **State space:** Upon receiving the KPI data at a certain step, OpenAI Gym interface formats them as a multi-dimensional bounded box to create a standardized state space that might encompass several slice-level metrics such as the number of new users, active users, computing resources allocated to each VNF, latency, energy consumption, number of VNF instantiations.
- **Action space:** On the other hand, the action space generally performs vertical up/down scaling of resources, i.e., increasing or decreasing the allocated resources to each slice in terms of e.g., VNF CPU and physical resource blocks (PRBs), as well as admits or rejects users within a slice according to the available resources. In this respect, at each step, the OpenAI Gym interface takes the DRL action as input and provides the next state observations, the action reward, and a *done* when the episode has been successful. Note that vertical scaling is generally limited by the amount of free computing resources available on the physical server hosting the virtual machine.

It is noteworthy that in a live network implementation, some APIs are required to either enforce the action sent by OpenAI Gym module into the network physical and virtual network functions or to read KPIs from them.

- **Reward:** This function should be defined carefully in the OpenAI Gym interface to guide the DRL agent towards maximizing slice admission rate and resource utilization while minimizing the total network cost, namely, CPU consumption, latency, energy consumption and SLA violations. In this intent, the reward is multi-objective, and can be defined as a weighted linear combination of the costs to be maximized (e.g., admission rate) and the inverse of those to be minimized (e.g. 1/energy, 1/VNF instantiation cost, 1/SLA violations). By tweaking the weights, the agent is guided to optimizing the costs that are prioritized by the slice tenant and/or the infrastructure operator.

DRL Control

The slice-level admission and resource control are based on continuous DRL algorithms where the main ones are summarized in the sequel and depicted in Figure 3-27.

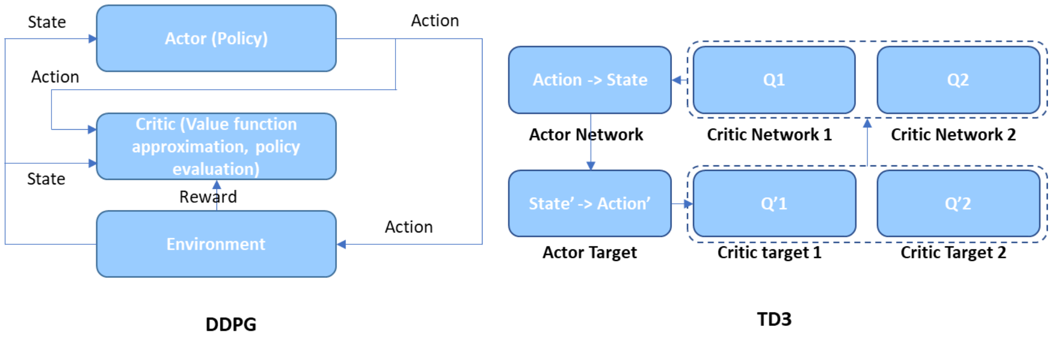


Figure 3-27 DPPG and TD3 diagrams

- **DDPG** [275] is a model-free actor-critic algorithm, combining policy gradient with deep Q-network (DQN). Indeed, DQN stabilizes the learning of Q-function by both sampling experiences form a replay buffer and freezing the target network (the network that estimates the Q-values is updated after several steps). The original DQN works in discrete space, and DDPG extends it to continuous space with the actor-critic framework while learning a deterministic policy. Instead of freezing target network---which delays the training---DDPG does soft updates on the neural networks' weights of both actor and critic via an autoregressive model that combines the old and new parameters in a smoother way.
- **TD3** [102], [107] introduces two changes into DDPG to sidestep the overestimation of the value function: **(i) Clipped Double Q-learning**: In Double Q-Learning, the action selection and Q-value estimation are performed by two networks separately. However, vanilla double Q-learning is sometimes ineffective if the target and current networks are too similar, e.g. with a slow-changing policy in an actor-critic framework. The *Clipped Double Q-learning* instead uses the minimum estimation among Q1 and Q2 on one hand and Q'1 and Q'2 on the on the hand, and leads thereby to a preference for states with low-variance value estimates, i.e., to safer policy updates with stable learning targets. **(ii) Delayed update of Target and Policy Networks**: TD3 updates the policy network at a lower frequency than the Q-function. This idea is similar to DQN; and **(iii) Target Policy Smoothing**: TD3 introduces a smoothing regularization on the value function by adding clipped random noises to the selected action and averaging over mini-batches.

3.3.4.4 Joint slice based demand prediction and resource allocation

Once network services are associated with a slice from the service layer and the orchestration, slices must be allocated sufficient resources. Due to the prevailing softwarization of mobile networks, such resources are mainly of computational nature. This holds both at the RAN where they map to, e.g. CPU time for containers running baseband units (BBU) in Cloud Radio Access Network (C-RAN) datacentres, and in the Core Network (CN) where, e.g., virtual machines run softwarized 5G Core (5GC) entities in datacentres. In this case, ensuring strong KPI guarantees often requires that computational resources are exclusively allocated to specific slices, and cannot be shared with others [145]. The dynamic allocation of network resources to the different admitted slices is, in fact, a chief management task in network slicing. In this context, the network operator needs to decide the amount of resources that should be dedicated to the different slices in advance, so as to ensure that the available capacity is used in the most efficient way possible and thus minimise operating expenses (OPEX). The key trade-off is between:

- Under-provisioning: if the operator allocates less capacity than that required to accommodate the demand, it incurs into violations of the SLA established with the tenant;

- Over-dimensioning: excess resources assigned to a slice imply a cost in terms of unnecessarily allocated resources that go unused.

Finding the correct operational point requires (*i*) predicting the future demand in each slice [63], and (*ii*) deciding what amount of resources is needed to serve such demand. These two problems are complex per-se. Forecasting future demands at service level requires designing dedicated, accurate predictors. On the other hand, allocating resources in a way that under-provisioning and over-dimensioning are poised to minimise the OPEX of the operator requires estimating the expected (negative and positive) error of the prediction. Moreover, addressing (*i*) and (*ii*) above as separate problems, risks to lead to largely suboptimal solutions, since legacy predictors do not provide reliable information about the expected error they will incur.

While the complexity of the complete solution may be daunting with traditional techniques, AI can be leveraged to address both aspects at once, by solving a capacity forecast problem. This can be realised by training a typical Convolutional Neural Network (CNN) architecture for time series prediction with a dedicated loss function that, instead of simply minimising the error, accounts for the respective costs of SLA violations and capacity over-provisioning, as done by the DeepCog algorithm [77].

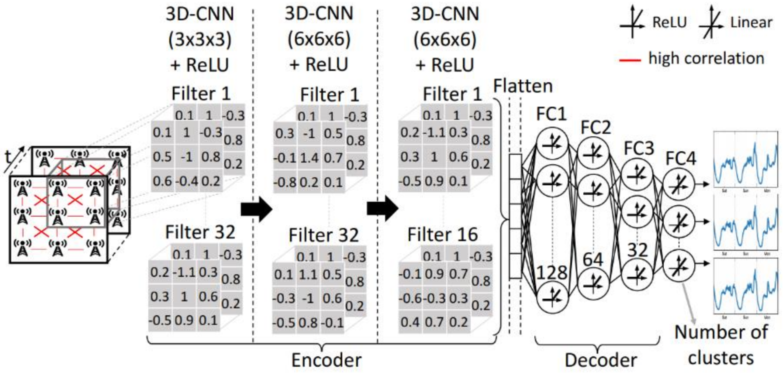


Figure 3-28 The DeepCog Neural Network Structure

Deepcog addresses the problem of optimizing the utilization of resources across network slices: measurement-driven studies show that, already under today's traffic demands, ensuring resource isolation among slices risks to yield unbearable costs for operators, as granting resources to slices under mildly efficient allocation strategies may require a six-fold increase of available capacity [72].

The core of the Deepcog algorithm is the loss function that translates the forecast load level into a feedback signal that proactively steers the network configuration to optimal levels. This loss function [76] is driven by the monetary value resulting from capacity allocation decisions: (*i*) the unnecessary provision of resources has a cost that proportionally increases with the amount of their unused quota, while (*ii*) negative errors (i.e., resources are not enough at a given point in time) have to yield a high economic penalty, irrespective of the error magnitude, to model an SLA violation. This effect is depicted in Figure 3-29 (left), where the Deepcog's loss function is represented as a function of the SLA violation cost βs and the costs resulting from the capacity overprovisioning γj.

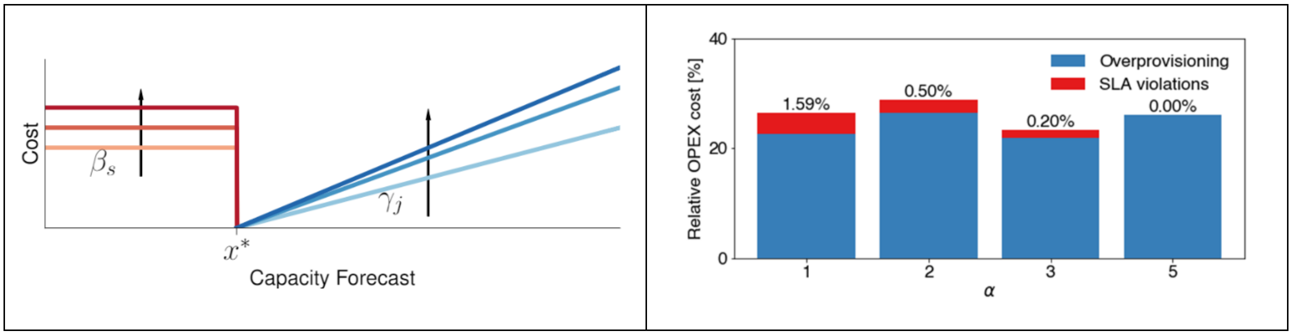


Figure 3-29 DeepCog cost function (left) and Overall result assessment (right)

The same Figure 3-29 shows the overall cost obtained by orchestrating slices with Deepcog for real-life measurements data as a function of α=βs/γj. Costs are expressed as the extra cost incurred (in %) over an oracle that can perfectly predict future demand. We observe that Deepcog is very effective, as the extra cost over the oracle is only around 20%. Furthermore, Deepcog is able to reduce the number of SLA violations as the relative cost of SLA violations (i.e., α) increases.

3.3.4.5 AI assisted slice isolation

Slice isolation here refers to the ability of including shared resources and services inside network slices that are associated with different application domains. Isolation can be applied in different parts of the network, such as the Cloud/Core, the transport network, the Radio Access Network (RAN) as well as at the UE level. For example, when focusing on the Cloud/Core, three isolation options are available, namely: 1) the fully shared, which is equivalent to the best-effort mode, 2) the partially shared, where most of NFs are shared and a few NFs independently deployed, and 3) the fully independent mode, where all NFs independently deployed. This is illustrated in the following Figure 3-30.

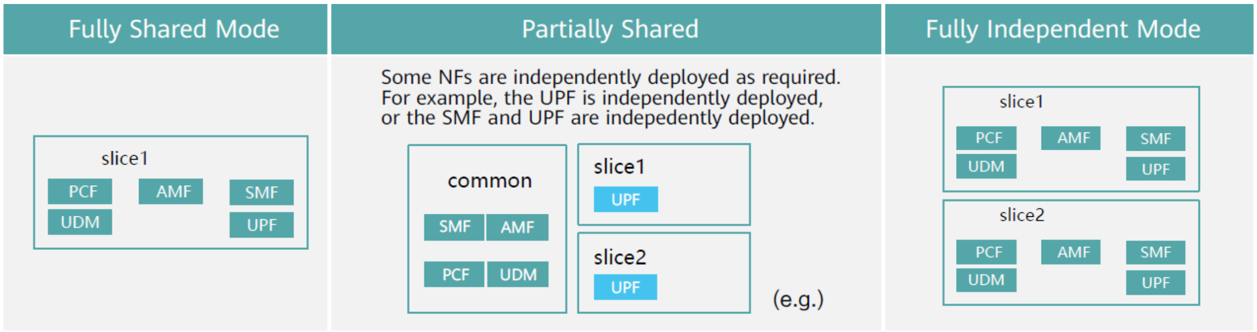


Figure 3-30 - Resource isolation mechanisms for a network slice

A challenge to be solved is how to provide automated mechanisms for deploying network slices with isolation options. These mechanisms can be implemented within components that can interact with different NFV MANO and edge PoPs. An example component is the Slice Orchestrator (i.e. manager) illustrated in Figure 3-31, which enables the (virtualized) network elements and functions to be easily logically segmented, configured and reused in isolation from one another. The Slice Orchestrator has interfaces for:

1. Checking the ownership and other requirements of a slice
2. Finding an optimal placement of all the resources involved in a network service
3. Setting up the slice components in the infrastructure through a VIM technology-independent module

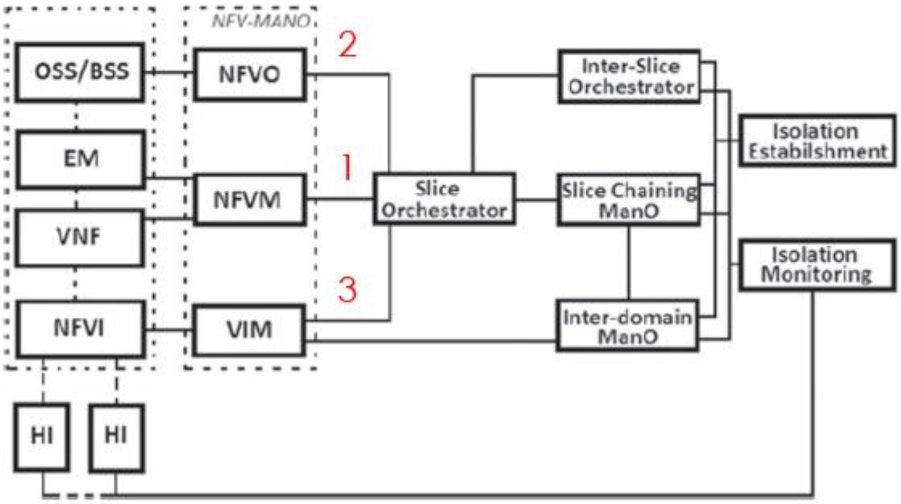


Figure 3-31 Slice Orchestrator interfaces for resource isolation

The Slice Orchestrator achieves complete isolation between resources as it interacts with the NFVI domain, where it creates a new tenant for every new slice to be deployed. Resource isolation can be further automated using ML techniques, such as deep neural networks that are described in Section 2.1.2.

Specifically, deep neural networks are used to predict the load of the network as well as of the individual resources belonging to each network slice based on previous information of incoming utilization and connections. This allows setting realistic thresholds for each network slice, such that there is no overlap in the employed resources or the network bandwidth. Prediction is based on three phases, including a profiling phase, a forward analysis phase, and a backward analysis phase. In the profiling phase, all network slice instances in the training dataset are fed into the model, each instance has a set of slice measurements and thresholds at each neuron. The thresholds of each neuron for all input network slices are logged, which is the output of the profiling phase. Then, in the forward analysis phase, the utilization and connection measurements for the resources of each network slice are fed into the model. Finally, in the backward analysis phase, we start from the output neurons defined and iteratively compute the resource contributions from the preceding synapses and neurons. The synapses and neurons with larger contributions belong to the network slices that are most prominent to exceed the allocated slice thresholds and need to be extended in order to guarantee resource isolation.

3.3.4.6 AI/ML-based decision making for slice optimisation

Network slices, in which services are offered to the users and verticals, are deployed and managed involving heterogeneous resources in the infrastructure. Hardware components (servers, sensors, antennas, UEs) are decoupled from software (SDN, virtualization,…) to maximize the use of resources in the infrastructure, which needs to be tightly evaluated and anticipated in order to have the overview and overall control of the network. The involvement of AI/ML techniques is a key enabler to comply with the SLA expectations and avoid system failures. The QoE is assessed from the end-user perspective, directly bounded to the QoS that the network is providing in the infrastructure. Network Services play a vital role to offer critical information of the usage and performance in the virtualized infrastructure, in that sense the service needs to provide the needed measurements that are ingested into a common database to analyse and evaluate the status and performance behaviour of the infrastructure. Analytics evaluation modules, based on the data sources retrieved from the service, can structure the metrics into relevant information to identify and improve the performance of the network and prevent errors. ML techniques, through forecasting events and obtaining knowledge of the behaviour, provide a predictive approach to achieve optimal slice decision to control network resource management and network functions involved in the service provision. Therefore, ML techniques offer additional support enabling the system to explore (big) data and deduce knowledge, this model-centric approach hides complexity in the network and simplifies and automate daily operations.

Analytics or ML models can also be used for optimising slices at runtime. There are several scenarios that can be addressed by runtime optimisations, including re-starting services that have unexpectedly stopped running, updating the slice placement when existing nodes/links in use become overloaded or when some of the nodes are under security threats. One way to achieve this is through the implementation of a Slice Optimisation Module, co-located with a Slice Manager as depicted in Figure 3-32. The Slice Optimisation module is based on a policy engine connected to an Analytics/ML model. A potential policy engine is APEX [42], which was also released open source as part of ONAP, but can be used independently from ONAP. APEX provides a strong tool for automated decision making, being able to handle adaptive policies, i.e., policies that can modify their behaviour based on system and network conditions, including decision making at run time rather than using pre-defined static policies. In the slicing optimization scenario described above, APEX accepts input events and requests from a Slice Monitoring component, routes the input to the appropriate policies, computes the policy results and generates response actions towards the Slice Manager to adjust the slice resources accordingly.

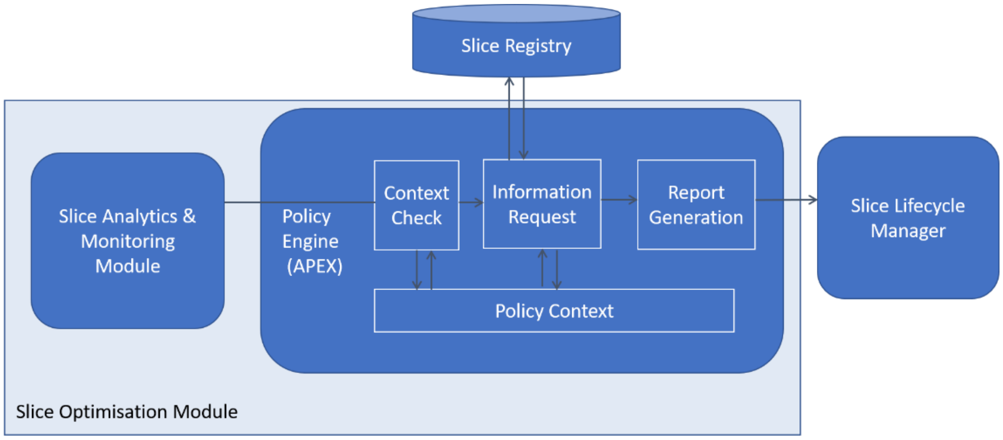


Figure 3-32. Example of using ML with a policy engine for slice optimisation

The Slice Monitoring & Analytics Module can include an ML model that predicts e.g., availability of infrastructure resources. ML methods offering time series prediction include ARIMA or LSTM, whose prediction result can be used for pro-active slice management at runtime. ML can also be combined within the APEX policy engine for deciding between the different available policy tasks – here a classification engine, or a Reinforcement Learning algorithm could be used to decide on the weights between the different policy tasks, etc.

The optimization of the resources thanks to the AI/ML techniques enables a better usage of the resources in the infrastructure. Models trained in the APEX system can be enhanced with the historical information that is stored in the dataset, gathered by the Monitoring tool (both infrastructure and performance) to enlarge the knowledge of the automated the decision-making tool.

3.3.5 Security

As discussed in Section 3.2.3 AI/ML techniques are already being applied to the detection of security incidents. Researchers though are investigating how to bring AI/ML techniques one step further, by not limiting their application to the detection phase, but also enabling these techniques to respond to the attack by acting on the network.

Table 3-11 includes two network security related use cases presented in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-11 Use cases for network security related aspects

Use Case	5GPPP Project	Additional references
Moving Target Defence for Network Slice Protection	INSPIRE-5GPLUS	[243]
Robust Self-Protection Against App-Layer DDoS Attacks	INSPIRE-5GPLUS	[56], [55]

3.3.5.1 Moving target defence for network slice protection

Classic static security approaches, such as firewalls, security protocols, authentication and encryption are necessary for the hardening and the safety of an ICT system, but they are not sufficient, as it is known that there is nothing such as a perfect and safe system. Malicious actors will eventually find vulnerabilities and exploits. As the attack surface greatly increased in 5G systems, securing the different verticals and the whole infrastructure becomes more challenging. Moving Target Defence (MTD) is a novel dynamic security approach performed on 5G verticals and slices, enabled by the virtualization of networks and network functions (SDN and NFV). MTD allows reducing the surface attack by dynamically changing different elements of the network, like its topology, the address space layout, IPs and port numbers, proxies, virtual machines and the instruction set. This capability makes the attacker's task more difficult, as when he tries to perform reconnaissance and fingerprinting to find a vulnerability, the MTD will change the properties of the 5G network slice resources, making the intelligence data gathered by the attacker obsolete. Apart from preventing attacks proactively, an MTD based approach can also act to mitigate one in an adaptive and efficient manner.

In our solution, the MTD framework shown in Figure 3-33 uses two components:

- MTD controller, **MOTDEC**, which is responsible for enforcing MTD actions
- An ML-driven Optimizer for Security Functions, namely **OptSFC**, that provides MTD strategies to maximize its attack prevention/mitigation, to minimize computational costs, and keep under control the QoS of the network services, keeping them under the requirements of the Security Service Level Agreement (SSLA).

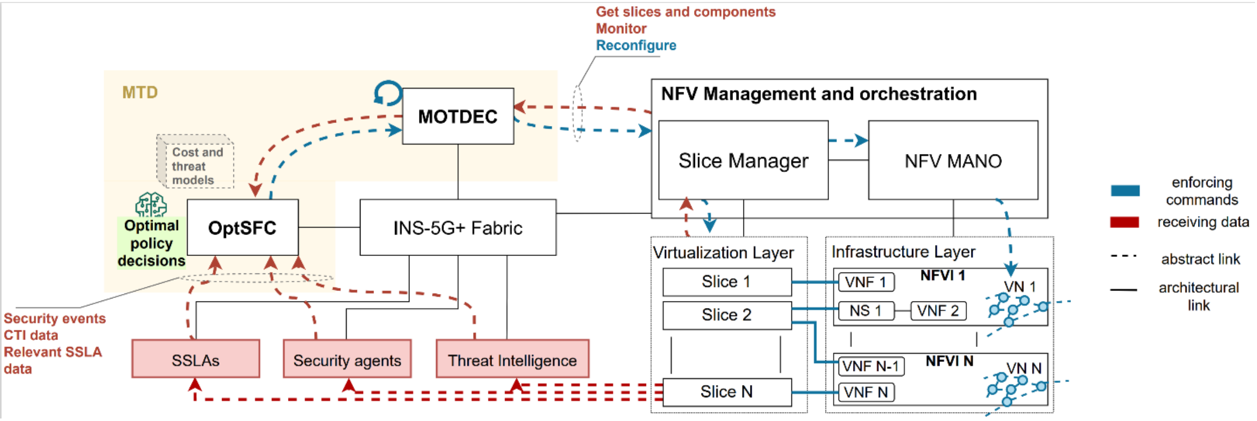


Figure 3-33. MTD for network slice protection

In other words, the MTD requires a cognitive system that dynamically determines what to move, where to move and how to move, based on the received input and on the action costs, in order to facilitate an optimal mitigation action. AI/ML techniques enable MTD to be trained on how to evaluate the cost of different security actions based on the actual state of the network and on the gravity of the threat (both of which are modeled and quantified during the learning process). For this purpose, OptSFC will model the state of the network using an incomplete information Markov game, as the defender does not know directly what the attacker is doing, but can perceive it through the network changes or the notification of a security agent. To this end, OptSFC is fed with real-time monitoring data provided from the NFV management and orchestration, as well as threat intelligence and anomaly detection alerts coming from security agents as shown in Figure 3-33. The Markov Decision Process (MDP) built will then be used for the training of a Reinforcement Learning (Deep Q-Learning DQL) agent, integrated in OptSFC and providing an MTD policy to the enforcer, MOTDEC. Therefore, to direct the MTD towards the optimal solution, our security approach relies on the usage of AI/ML, which allows the system to continuously learn and optimize its actions.

3.3.5.2 Robust self-protection against app-layer DDoS attacks

The expected high bandwidth of 5G and the envisioned massive number of connected devices will open the door to increased and sophisticated attacks, such as application-layer DDoS attacks. Application-layer DDoS attacks are complex to detect and mitigate due to their stealthy nature and their ability to mimic genuine behaviour. To tackle this issue, a robust application-layer DDoS self-protection framework that empowers a fully autonomous detection and mitigation of the application-layer DDoS attacks leveraging on Deep Learning (DL) and SDN enablers can be used. The DL models have been proven vulnerable to adversarial attacks [54], which aim to fool the DL model into taking wrong decision. To overcome this issue, the DL-based application-layer DDoS detection model is made robust to adversarial examples.

Figure 3-34 depicts the basic architectural components of the proposed solution to mitigate the application-layer DDoS attacks in a fully autonomous way. The “App-Layer DDoS Protection” component is in charge of detecting the malicious activity and issuing the security policy in case the attack is detected. It consists of four main modules: the “Network Flow Collector”, the “Features Extractor”, the “Detector” and the “Security Orchestration Plane”. The Network Flow Collector permanently collects network flows via port mirroring. To limit the impact of mirroring on the net- work performance, only traffic flowing from/to the monitored asset (e.g., Web server) is mirrored. The collected traffic is periodically exported to Features Extractor to retrieve flow’s features relevant to application-layer DDoS attack detection. Once extracted, the network flow features are passed to the Detector for uncovering suspicious behavior. The detection is performed by DL model built using Multi-Layer Perceptron (MLP) algorithm. The proposed model consists of 1 input layer, 2 hidden layers with 64 neurons each, and a two-class softmax output layer. The model's input is the flow features received from the Extractor, while its output is the traffic class; that is, DDoS traffic or legitimate traffic. If a malicious traffic pattern is identified, the Detector issues a security policy (e.g., flow dropping or steering) to the Security Orchestration Plane. Upon receiving the security policy, the Security Orchestration Plane converts the policy into a flow command and sends it to the SDN controller. Based on the received flow command, a flow rule is pushed by the SDN controller to the corresponding virtual Switch (vSwitch) to fulfil the defined security policy.

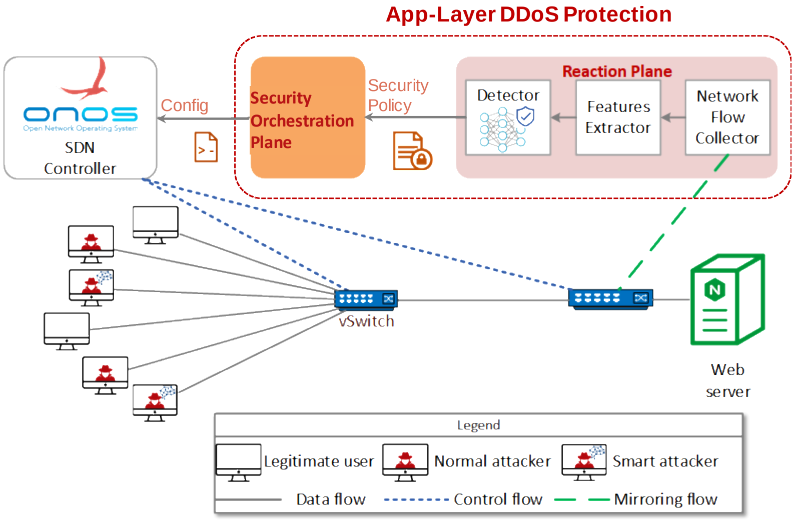


Figure 3-34. The Robust App-Layer DDOS Self-Protection Framework's High- Level Architecture

The framework's robustness stems from its ability to mitigate adversarially-generated App-layer DDoS attack flows; that is attack flows generated in a way to evade detection by a ML-based detector. To this end, the adversarial training defence [55] is adopted to make the MLP model robust to white-box evasion attacks. In adversarial training, the MLP model is explicitly trained on adversarial examples in order to learn how to resist them. The white-box attack Fast Gradient Sign Method (FGSM) [129] is considered for the purpose of this work. FGSM generates adversarial examples by performing a one-step gradient update in the direction of the gradient's sign of the loss function relative to the input. The input is than altered by adding a perturbation that can be expressed as: $\eta = \epsilon . sign\left(\nabla_x J(\theta, x, y)\right)$, where x is a sample (i.e., network flow), y is label of x, $J(\theta, x, y)$ is the loss function used to generate the adversarial example and ϵ is the perturbation magnitude [55].

The devised framework is implemented and deployed on an experimental testbed [56]. The performance results show the effectiveness of the proposed framework in protecting against application-layer DDoS attacks even in presence of adversarial attacks.

3.3.6 Application functions

The integration of the application domain and the mobile network domain has become a critical aspect in 5G networks, which target the adoption of this technology in vertical domains like automotive or manufacturing. Supporting verticals over 5G networks has been one of the key areas of interest of the 5GPPP community [10]. In this section we provide example about how AI/ML techniques can be applied to support application functions in 5G networks.

Table 3-12 describes two application function related AI/ML use cases presented in this section, while providing the interested reader with additional resources developed in the corresponding 5GPPP projects.

Table 3-12 Use cases for application function related aspects

Use Case	5GPPP Project	Additional references
Dash prefetching optimization	5GZORRO	[169], [336]
Q-Learning application to federated scenarios	5GROWTH	[29], [42]

3.3.6.1 Dash prefetching optimization

Video content is the dominant traffic on the Internet due to content popularity and the high bitrate required compared to other mobile applications. Cisco anticipates that video content will constitute more than half of the traffic crossing mobile networks by 2022 [62]. With the ongoing deployment of 5G networks, the expectation for high-quality 4K/8K video streaming over mobile networks has also raised. The emerging MEC technology allows MNOs to provide services at the edge of the network. Considering the scarcity of physical resources at the edge and high storage demand of video content, effective management of video traffic at the edge by MNOs will become increasingly critical to satisfying end-users with the promised QoE as they stream higher volumes at higher bitrates.

Currently, dynamic adaptive streaming over HTTP (DASH) [198] is the dominant video delivery standard and has been adopted by most content providers like YouTube and Netflix [52]. Videos in DASH are split into equally sized segments available at multiple video bitrates. The dynamic nature of mobile networks necessitates adjusting video bitrate based on the network and playback buffer's status. The process of requesting a video segment is performed by the adaptive bitrate (ABR) algorithm, which utilizes monitoring data and adjusts the next segment request's bitrate to maintain the highest achievable QoE for the users.

Caching video content closer to the users, at the mobile edge nodes, yields benefits both for the users and the MNOs. It decreases users' content access delay, improving their QoE while also mitigating the load on the backhaul link for the MNOs. Nevertheless, the mobile edge nodes' limited capacity calls for intelligent decisions on what content to cache and where to cache it to improve QoE while also efficiently using the network resources. Despite the notable advantages of in-network caching, it cannot be a practical solution for DASH video streaming in the context of mobile networks. First, the number of users served from a gNB is usually relatively small, and since segments are downloaded in sequence, caching a segment might not be re-used by many of the users served from the same gNB. Therefore, it is likely to see a low cache hit ratio for most of the cached segments. Second, caching proves its usefulness when the cached segment is used more than once. In other words, in any case, the first user that requests a segment will be served from the main video server, and the user will experience a poor QoE.

Therefore, novel solutions must address this issue of pre-fetching video segments to the edge before the segment is requested. Content pre-fetching is a technique to anticipate the requests and move the contents close to the end-user based on the predictions, aiming to reduce the time for delivering the content to the user and avoid excessive backhaul bandwidth utilization. Pre-fetching requires being performed intelligently to avoid excessive pre-fetching of video contents that are less probable to be requested by the users, resulting in extra storage and bandwidth utilization [169]. In this regard, prediction, anticipatory pre-fetching, and caching of video segments of the right bitrate during streaming at the mobile edge nodes play a pivotal role in MEC-enabled DASH video streaming. MEC's services (e.g., RNIS) can be greatly used by the system to make proper predictions. Moreover, novel techniques are required to efficiently employ the knowledge produced by predicting user behavior to pre-fetch the contents.

Performing predictions in radio access networks (RANs) is especially challenging due to frequent changes in physical channel conditions and the availability of different radio access technologies. Employing ML to predict specific metrics (e.g., channel throughput) for RAN has gained importance in the past years. Within the context of DASH, the state-of-the-art ML works have mainly focused on bandwidth estimation at the client, which constitutes an input to most ABR algorithms as well as predicting several parameters that the ABR algorithm uses for requesting future segments (e.g., client bandwidth, buffer, and bitrate of previously requested segments). Supervised learning algorithms such as random forest, extreme gradient boosting, support vector machine, and neural networks are examples of ML algorithms that have been extensively applied to this problem.

3.3.6.2 Q-Learning application to federated scenarios

The 5G networks require the concept of federation to orchestrate network services across multiple administrative domains. In such context, the consumer domain, which is the domain managing the whole orchestration process, needs to deploy part of the service to an external provider domain. For each requested service from the vertical users, the consumer domain needs to decide how to deploy the service. There are three deployment options: (*i*) to instantiate locally, (*ii*) federate or (*iii*) reject a vertical request. For each decision generated, the goal is to satisfy the vertical requirements and maximize the profit of the consumer domain.

Greedy heuristics strategies [29] show that the decision can affect the profit outcome. Applying a reinforcement learning strategy, such as Q-learning, can generate more profitable deployment decisions in a federated ecosystem.

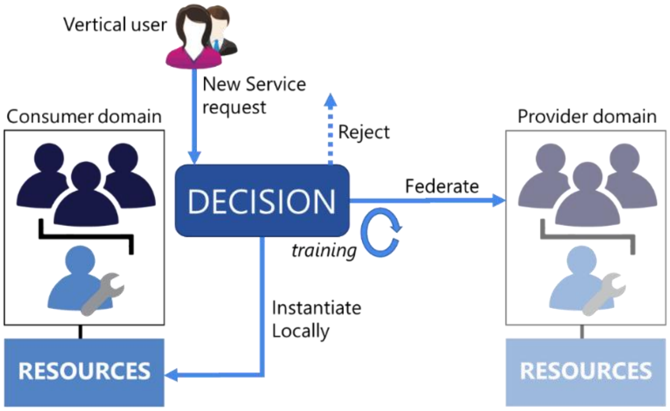


Figure 3-35 A Q-learning decision scenario to maximize operator's revenue

The work in [41] depicts the scenario of Figure 3-35, where a Q-learning agent is applied to generate decisions in the consumer domain. The idea in [41] is to train the Q-table agent through trial and error (of the deployment options: local, federated, reject), and to learn to generate more profitable decisions. For that, the agent is endorsed for every positive decision and penalized for every wrong decision (e.g., deploying locally a service that exceeds the local capacity). The results of the Q-learning cumulative profit are compared against a greedy, non-federating approaches and the optimal revenue for a given set of vertical requests. Although the results show near optimal profitable income, the future application of deep learning approach should increase the revenue performance.

4 Architectural aspects

This section discusses the impact of AI/ML in network architectures. It presents specific architectural solutions introduced in 5G PPP projects, and discusses the requirements for ML model lifecycle and interface management. The chapter includes a reference to global activities, either in the context of standardization organizations or open-source initiatives. It discusses trust in AI/ML-based networks and provides a brief overview of AI/ML-based KPI validation and system troubleshooting.

4.1 Application of AI/ML in a network architecture: 5G PPP solutions

This section provides several 5G PPP solutions that requiring the deployment of dedicated frameworks for the support of AI/ML solutions. These solutions require appropriate modifications in the overall architecture. As there are global activities that also provide their approaches for introducing AI/ML in the networks, the last subsections maps the 5G PPP solutions to the global activities.

4.1.1 AI-based autonomous slice management, control and orchestration

AI based 5G autonomous network slice management, control and orchestration is achieved through an innovative architecture based on an enhanced Monitor-Analyse-Plan-Execute-Knowledge (MAPE-K) loop, as shown in Figure 4-1 [247]. Firstly, the Monitoring Sub-plane gathers concerned metrics and other contextual information such as metadata from the infrastructure especially the underlying 4G/5G network data and control planes regarding network and cloud resources, user traffic flows and network topology. These monitoring data are fed into a Data Lake in the Information Sub-plane, and the Data Lake processes the data through aggregation and Big Data analytics. Then the analytic outputs are sent to the Cognition Sub-plane, where further AI/ML-based analyses are conducted with further aggregation from other sources especially feedback from vertical end users including User Equipment and subjective quality evaluation of their services via the One-Stop API (OSA) and Plug & Play Control. The outcomes are employed to either update existing policies/rules or create new ones regarding network management, control and orchestration tasks, with a focus on network slicing, by the Policy Framework. These policies are then implemented by the Orchestration Sub-plane, which orchestrates the required actions based on the policies and the required resources for such actions. The orchestration is performed both vertically across resources, slices and services and horizontally across various Network Service Provider (NSP) domains involved in delivering the services. Quality of Experience (QoE) and FCAPS management is also in place in this sub-plane to ensure the performance of the system. The actions orchestrated are executed through the Control Plane, which is an overlay plane on top of 4G/5G and beyond network infrastructures comprising data and control planes in themselves. The Control Plane consists of a set of adapters and controllers specific to each network segment within each domain, covering RAN, MEC, backhaul, data plane programmability (DPP) for all the links between segments, and wired area network for inter-domain slicing-friendly communication. These adapters and controllers ensure the actions are executed in the right segment and in a technology-agnostic manner to be applicable over heterogeneous underlying network infrastructure technologies. After the execution of the actions, the monitoring process regarding the new network and service status continues the next-round loop and closes this loop. Through the above closed-loop operations, autonomous network management, control and orchestration are achieved. Based on the above architecture and procedures, two ITU-T Draft Recommendations have been created. One recommendation [83] focuses on ML-based network slice management and orchestration across multiple NSP domains. The other one [84] emphasises the schemes to allow vertical users to influence the run-time optimisation of their network slices. ML algorithms for this purpose can be on boarded on demand at runtime through the P&P control framework developed and demonstrated in [245].

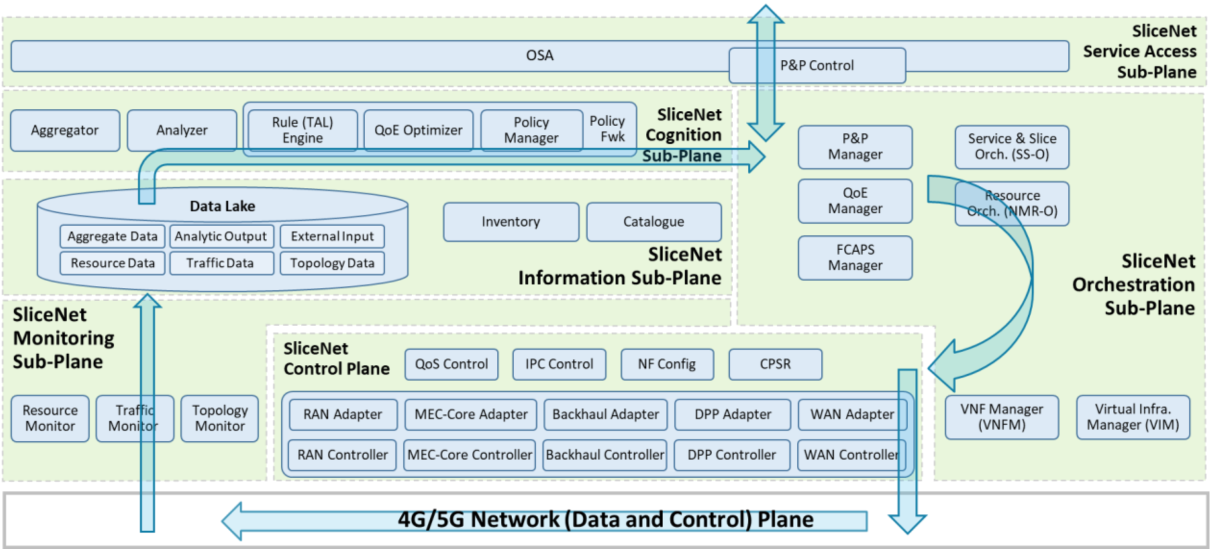


Figure 4-1: Overall architecture for AI based autonomous management, control and orchestration

4.1.2 AI/ML-based scaling operations in a network service orchestration platform

The autonomous assurance of service level agreements (SLA) is the next big challenge in 5G networks to foster the advantages introduced by NFV/SDN paradigms to automate the network management. For that, management and Orchestration (MANO) platforms can rely on AI/ML techniques to automate the triggering of network service scaling operations aimed to re-establish the requested SLA. The E2E vertical service platform designed and developed in [238], based on the ETSI NFV architecture and specifications, integrates such techniques also following some of the architectural concepts proposed by O-RAN, still under definition. According to such principles, the architecture in [238] defines the AIML platform (AIMLP) to perform model training [58], while other building blocks, like its Service Orchestrator (SO) or its Resource Layer (RL), oversee the inference task.

To carry on their AI/ML activities, the AIMLP, the SO or the RL rely on i) the available Vertical Oriented monitoring system (VoMS), which is the component in the architecture in charge of collecting monitoring data from deployed network services and NFVI; and ii) the integrated data engineering pipeline in charge of ingest, process and analyse the data. The data engineering pipeline included in the platform relies on well-known open-source tools, namely, i) Apache Kafka to ingest the data used to perform AI/ML decisions, ii) Apache Spark to generate models from training data and to perform inference, and iii) Apache LIVY as a REST-API to interact with Apache Spark to submit/terminate the different training or inference jobs. The fact of using such open-source tools provides the platform with reliability and the access to a great variety of AI/ML techniques.

Though the framework can handle any AIML-based problem, in the case of AIML-based scaling operations, the SO, in charge of the lifecycle management of network services, has been evolved to coordinate the process thanks to the existence of a new proposed information element extending the ETSI NFV-IFA 014 NSD template. This new IE expresses the need of interaction with the AIMLP to configure AI/ML-based decisions for a given MANO problem (in this case “scaling”) and specifies the metrics out of the ones already defined for this kind of network service in the NSD field “monitored Info” required by this AI/ML problem to perform its decisions. Based on some contextual information (e.g., initial instantiation level) and the required monitoring information, the SO launches an inference job which will decide the best instantiation level in the current network conditions, triggering the scaling operation if the decided instantiation level does not coincide with the current instantiation level.

A complete description of the architectural evolution of the SO and its operational workflow to support the scaling-based operation is presented in [132]. This architectural evolution shows the required interaction with the AIMLP to obtain the corresponding model and the inference file and the inclusion of the open-source tools to run the inference process triggering the possible scaling operation. Additionally, [132] presents an experimental evaluation of the additional operations required to configure the AI/ML-based scaling process during different instants of the network service lifecycle management, namely instantiation, run-time, and termination. This evaluation shows that AI/ML-related service handling operations (1-2 s.) are well below instantiation/termination procedures (80/60 s., respectively). Furthermore, online classification, which in this evaluation is performed with a random forest classifier, can be performed in the order of hundreds of milliseconds (600 ms). Currently, the most time-consuming operations are the ones involving interaction with the Apache Kafka open-source tool to publish/consume the data. This work sets the foundations for further study on the use of different AI/ML techniques to derive models for the scaling problem in a real experimental system and see, for instance, its impact in the inference time.

In terms of AI algorithms, more advanced AI/ML techniques can be explored to make smarter scaling decision to handle the traffic dynamicity and resource utilization. On the one hand, under-provision of the necessary computing and network resources will cause a network inability to support the incoming traffic loads and will thereby affect service performance; on the other hand, resource over provisioning can result in idle VM instances and therefore an avoidable cost for the network operator. Taking this fact into consideration -and since service workload is constantly changing-, achieving automatic up- or down-scaling to respond to the dynamic service requests is challenging. To this end, a proactive Machine Learning (ML) classifier and especially a Multilayer Perceptron (MLP) is proposed in order to proactively make scaling decisions based on (near) real-time network traffic statistics. The VNF auto-scaling problem is a supervised ML classification problem since in order to train the MLP model, a dataset including seasonal/spatial behaviour of network traffic load as features as well as previous VNF scaling decisions is used to generate scaling decisions ahead of time. Those features include traffic measured at time τ, traffic change in a period of time, average number of user connected in each cell in a period of time, etc. The classification output is the number of VNF instances required to accommodate future traffic loads without violating QoS requirements and deploying unnecessary VNF instances saving significant costs for the network owners as well as leasers.

In addition to the problem of inefficient resource utilization and traffic dynamicity, network performance may be significantly impaired when several independent management functions take actions on shared computing or network resources to optimize individual objectives. For example, to efficiently provision resources (CPU/memory), uncoordinated auto-scaling actions of concurrently operating management loops result in conflict events when resources are constrained, thereby leading to SLA violations and worse, suboptimal network performance. While a centralized auto-scaling algorithm would ideally not result in conflicting actions, a central orchestrator would be inefficient in detecting and resolving such conflicts during run-time when management loops need to operate in the order of milliseconds or less. Therefore, a decentralized solution framework is proposed that encourages management functions to take mutually cooperative actions optimizing performance from a network-wide view. To address these challenges, Multi-Agent Reinforcement Learning (MARL) [53], a sub-branch of RL is proposed in which several agents learn from their interactions with the environment. Here the management functions are modelled as agents. Although usual drawbacks for such an approach such as the curse of dimensionality and non-stationarity of the environment persists, smart techniques to include only relevant information about neighbouring agents, such as in [142] reduce the state space and learning within the multi-agent environment. By allowing the agents to learn the dynamics of the environment during the exploration phase, they achieve conflict avoidance behaviour, thereby reaching **close to optimal** performance.

4.1.3 AI/ML as a service in network management and orchestration

An AIML as a Service (AIMLaaS) platform has been designed that allows exploiting AI/ML models for the various decision points embedded in a 5G management and orchestration stack [238]. Examples of relevant decisions that would benefit from the AIMLaaS platform include network slice arbitration in the Vertical Slicer component, NFV network service deployment and federated provider selection in the Service Orchestrator component, path re-computation at the Resource Layer component, as well as any SLA management-related algorithm at any of the above layers (e.g., to decide on NFV network service scaling).

The models can be uploaded to the AIMLaaS platform by any authorized external user. Such models can be already trained and inserted simply along the necessary file to then perform inference, or to-be-trained models. In the latter case, the user can provide a suitable data set to be exploited for the training phase. In both cases, the user can specify (i) the *scope* of the model, i.e., the type of decision to be used for, such as service scaling or service arbitration, and (ii) the *type of service* the model/dataset should be used for, such as digital twin or video provisioning. When the external user uploads to the platform a to-be-trained model, it is the platform itself that takes care of the training and records the corresponding timestamp and, potentially, a validity time lapse. If no dataset is uploaded along with the to-be-trained model or whenever appropriate, the AIMLaaS platform can exploit the data collected through the monitoring platform about network/computing resource utilization or performance, or vertical service target KPIs. The configuration of the monitoring platform to gather the monitored data, its aggregation (e.g., through Kafka), and their feeding as input for real-time model execution in the corresponding building block also need to be properly set up.

Models stored in the AIMLaaS platform can be accessed by an architecture entity through a REST interface, by expressing the aforementioned scope and type of service. In this sense, an efficient interaction for model discovery, selection, and delivery must be defined. Furthermore, it is envisioned that an accuracy performance, and when training is executed, latency performance should be derived and provided by the AIMLaaS platform to the entity requesting a suitable model. Interestingly, a publish-subscribe paradigm could be also be implemented, in order for other entities to receive up-to-date models by the AIMLaaS platform.

The AIMLaaS platform is being integrated in the regular project’s service lifecycle management workflows, including its interaction with the monitoring platform [238]. In this way, the MANO stack, the AIMLaaS platform and the data engineering pipeline are being integrated towards automated and efficient network management.

4.1.4 Architecture aspects to enable ML in the verticals' domain

Although 5G networks can equip their network operations and management with internal AI functions, it is unavoidable to allow external parties for offering AI applications and services to empower the analytics capability of the network for two reasons. First, it is costly and resource-consuming to run powerful and advanced AI/ML algorithms inside the network, especially when the AI-driven management services are not always required by customers. Second, it is hardly possible to include all AI-driven services required by each customer inside the network, especially when the customers' requirements are highly dynamic. Therefore, it is more realistic to expose certain network data and management services to vertical customers and applications. Doing so could achieve better customer-oriented service performance. However, it also brings a challenge on network stability as sometimes the customer's intervention may cause confusion, conflicts or even failures. One practical and effective way is to expose network monitoring functionality and allow customers to run their own advanced AI/ML mechanisms to solve certain performance optimization problems. The produced recommendations can be considered by network operators. In [194], four exposure levels are considered, from the infrastructure and NF level to the NS and E2E service level. Lower levels of monitoring exposure (Level 1 and 2) offer limited datasets and abstract knowledge about the network conditions, which is mainly suitable for reactive analytics such as logistic regression. As a consequence, external AI is expected to be more proactive and advanced, e.g., deep reinforcement learning or long-short term memory cells neural nets (LSTMs). Higher levels of Monitoring exposure (Level 4 and 3) allow external parties for deeper network management and provide more detailed knowledge about the network conditions, including infrastructure and elements, which encourage proactive analytics built-in inside the network. Then external AI could run relatively simpler reactive analytics to complement the internal AI inside the network. In a word, external AI functions are complementary to the network built-in AI. With an increase in the exposure level, the analysis complexity is shifted from the customer's side to the network's side. Proper design and analysis is needed to allocate the AI responsibility between network operators and customers. In addition, Intelligence is needed to mediate and evaluate the recommendations from multiple customers towards the same network operations or network elements and make the best decisions [194].

4.1.5 Cross-layer optimization framework using ML

In this section, a framework for Autonomic Network Management (ANM) and network reconfiguration combining Software Defined Networks (SDN) with Software Defined Radio (SDR) via Network Function Virtualization (NFV) enabled Virtual Utility Functions (VUFs) is introduced [272]. The objectives of the proposed Autonomic Network Management Optimization in SDR environments (ANMO-SDR) architecture, include reconfiguration flexibility, efficient use of the bandwidth, as well as, efficient and transparent Device-to-Device (D2D) communications, without interrupting the primary network operation. The framework exhibits the Monitor-Analyse-Plan-Execute (MAPE) closed-loop function, implementing cognition through a learning model, in which past interactions with the environment guide current and future interactions, resulting in intelligent enhancements [271]. The proposed framework focuses on the three lower layers of the protocol stack, namely the Physical, Medium Access Control and Network layers, which are extended to include a vertical cross-layer interconnection. The entire stack is realized individually in each Mobile Node (MN) [148]. Specifically, the physical layer is enhanced with SDR features such as spectrum sensing.

The main functionalities are implemented in a cross-layer Resource Channel Allocation (RCA) mechanism for Cognitive Radio Networks (CRNs) that combines a Markov Random Field formulation with Gibbs sampling and allows the distributed and efficient operation of each secondary user. According to this, each MN seeks to "minimize its energy" by minimizing the cumulative neighbourhood energy function comprised of the sum of its singleton and pairwise doubleton potentials [148]. The state of each device depends only on the states and the information of its neighbours. Gibbs sampling can be applied by each device individually, reaching global optima through local sampling. Cumulatively, this distributed sampling converges to global optimizers of the system. Localized interference information is incorporated into the RCA mechanism via the inclusion of the cost vector in the second order potential function, which is constructed and maintained by a reinforcement learning algorithm, i.e., by adding a predetermined penalty (award) after every unsuccessful (collision-free) transmission to the respective channel component.

The cross-layer functionality related with the involved entities and the respective Autonomic Control Loop (ACL) are presented in Figure 4-2. Complying with the SDN concept, the proposed framework allows the decoupling of the control and data plane, where the latter implements only the data forwarding related operations and does not perform any type of control. As far as non-control functionalities are concerned, the three layers remain essentially unaltered compared to the traditional protocol stack. Specifically, the data plane of each MN is responsible for providing the ambient information of each individual layer to build self- and environment-awareness, through the information collection component. This cross-layer information from the data plane includes information about the gains of the wireless channels, the interference among the communication links and various network parameters such as queue lengths or the number of active traffic flows.

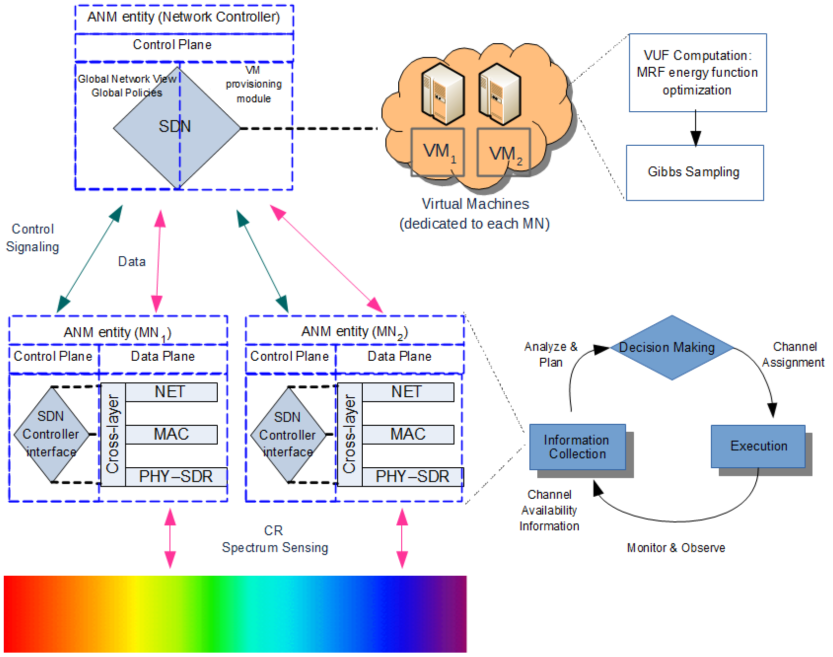


Figure 4-2: NMO-SDR cross-layer functionality [272]

Contrary to the data plane, the control plane is responsible for the decision making component, performing the RCA. The control plane utilizes the information collected by the information collection component and then decides the execution of the respective action, implemented by the execution component, returning feedback to the three layers (i.e., transmission channels, power level, etc.). In particular, the control plane of each MN, through the SDN controller interface, interacts with the control plane of the SDN controller by exchanging signalling messages. Accordingly, the SDN controller is responsible to arrange virtual resources to compute the Virtual Utility Function (VUF) related with the selection of the most efficient channel allocation, as requested by each MN.

Subsequently, the VM, dedicated to each MN, sends through the SDN-controller the relevant computation instance of the VUF to the requesting MN, enabling it to assign available channels according to its desired operational requirements. The chosen channel by each secondary user is communicated to the neighbouring secondary users, which update their status and sequentially proceed to the VUF computation, as a recurring process. The decision making is further enhanced by the SDN controller, which enforces global policies (i.e., “globally minimize collisions”, or enforcing a faster “best effort” channel allocation) regarding resource allocation for all three layers.

The reinforcement learning algorithm is designed with the purpose of increasing the spectrum utilization by minimizing the number of non-assigned available channels at each secondary user. To quantify the desire for increased bandwidth and fulfilled QoS requirements that drives each node to demand more radio resources (channels), a sigmoid utility function is employed, which depending on the parametrization can be used for different services with diverse objectives, based on the policies of the SDN controller. The potential function of doubleton cliques expresses the energy cost inflicted by the interaction of competitive secondary users and seeks for a joint optimization of physical-MAC layer operation. Through the assignment of orthogonal vectors between competitive users, collision-free secondary transmissions are guaranteed and channel congestion is avoided. Thus, the achieved spectrum utilization addresses the primary goal of CRNs (i.e., increase of spectrum utilization without interfering primary users. Additionally, the approach promotes the fair allocation of resources, since no particular node is constantly assigned more channels in comparison to the others.

4.1.6 Third party ML analytics for network operation optimization

This contribution targets one of the challenges in the deployment of end-to-end services and applications: the lack of visibility between the network’s domain and the domain of the vertical service. As a consequence, the relationship between network KPIs collected from the networks and the application KPIs measured by customers is not known and the impact on each other cannot be easily formulated. As such, the customer might see effects of network events that it is not aware of, and the network might become over-/under-loaded without knowledge of changes in the service. The idea is to run third party ML analytics to optimize service performance and potentially network operations. Once vertical use case services are deployed and activated, a set of network KPIs are collected and sent to an ML engine, together with network operation metadata (e.g., slice status, etc.) [240]. In addition, the customers themselves can measure the KPIs of their applications, which will also be sent to the ML engine. Using the ML solution for correlation, the customers aim to learn the relationship between these two seemingly different sets of KPIs and also enable predictive optimization of network and service performance.

The learnings on the correlation enable customers to gain a better understanding of how their applications perform under certain network conditions. As a result, customers’ analytics can provide alarms, predictions, and/or recommendations to network operators once certain performance issues are detected from their side. In this way, network operations will be adjusted based on requests from customers, which makes the network operations more customer-oriented or QoE-driven. The following ML-based analytic solutions are envisioned [240]:

- **Correlation and Clustering:** One way to find the relationship between KPIs is to detect similarities between KPI trends based on their behaviour over time. This can potentially be used for root cause analysis and determining whether a change in a KPI causes a performance issue in another. Further, by finding groups of KPIs that have similar temporal behaviour, monitoring can be scaled down to single KPIs from a group of strongly correlated KPIs rather than continuously monitoring all KPIs. Some of the methods to find similarities between time series include correlation (Pearson’s, Kendall’s, etc.) and Dynamic Time Warping. Once correlations between KPIs are calculated, they can be used to cluster KPIs that share similar behaviour over time using algorithms such as Hierarchical or Graph clustering. It is important to note that correlations, while indicating similarities between time series, do not necessarily indicate a causation and that needs to be further investigated by the application owners.

- **Anomaly Detection:** KPIs collected from different domains can sometimes suffer deviations from expected values that can indicate issues in the network and/or a drop in performance in the vertical service. Anomaly detection of time series enables the system to determine when monitored KPIs are out of expected range by a comparison to their historical values. Some of the methods used for anomaly detection include Median Absolute Deviation (MAD), Auto-Regressive Integrated Moving Average (ARIMA) and recurrent neural networks such as Long Short-Term Memory (LSTM).
- **Prediction:** Since KPIs from different domains can have a direct impact on each other, this relationship can be used to predict future values of KPIs based on historical values of other monitored KPIs. Further, prediction can also be used in automation, e.g. a predicted out of bounds value in the next interval could trigger a pro-active, corrective action. The ML algorithms used for prediction of time series data vary from classic algorithms such as decision trees and random forests to newer approaches based on artificial neural networks (ANNs) such as Multi-Layer Perceptions (MLPs) and Convolutional Neural Networks (CNNs).

4.1.7 AI-based anomaly detection for 5G and beyond networks

This section analyses and identifies new requirements, features, enablers and adaptations needed to integrate AI-based anomaly detection systems on multi-tenant and multi-domain 5G networks. Current AI-based solutions for cyberattacks detection such as [211], are not yet fully adapted to work with 5G traffic, in particular requirements raising from envisioned real-life 5G networks: 1) The AI-based anomaly detection system should support handling **multitenant and multi-domain** network traffic, so that the network can be exploited concurrently by different kind of verticals. 2) **Scalability**, as the AI-based system should be able to handle millions of data coming from different devices, and network elements, thereby supporting **mMTC** scenarios on real-life scenarios such as ambient monitoring in smart-agriculture. 3) The AI-based system should be able to consider the real-time data-path analysis under very demanding network conditions rising from different 5G services requirements, **eMBB** services for broadband-communications e.g. video surveillance in smart-cities, **URLLC** services with low-latency requirements, e.g. Robot control in industrial automation. 4) The AI system should be self-adaptable with reinforcement capabilities to consider dynamically additional features in the analysis. 5) The AI system should be supporting **efficiently** the management and control of heterogeneous network traffic. 6) Flexible and adaptive management of the AI system itself. 7) **Interoperability** in the AI management with common data-models with different kind of features, datasets, attacks/threats and heterogeneous kinds of network traffic. 8) Fully **distributed**, the AI system should be run in a fully distributed way and should consider the privacy-preserving issues in federated learning, where different verticals and operators might collaborate to come up with enhanced and scalable detection AI systems and models.

To cope with those requirements, a novel AI-based virtual network functions and associated monitoring agents can be automatically orchestrated, delivered and enforced on demand as virtual network functions, to perform deep packet inspection on real time on the data path identifying flows and meaningful features and then, detect, in a distributed way, anomalies on the 5G network traffic that might occur at any network segment of the multi-tenant, multi-domain 5G-IoT networks.

The system should be endowed with a monitoring and deep packet inspection module that will work exclusively with 5G multi-tenant traffic. Due to the inclusion of new protocols to make the network virtualization in tenants possible, as well as to continue guaranteeing user mobility, 5G multi-tenant packets will appear encapsulated with protocols such as GTP and VXLAN, so our module must be able to parse packets with this structure and extract metadata correctly. This parsing will be done at a very low level by handling manually defined data structures and offsets, in order to obtain quite competent performance in execution times, which is necessary for the rest of the system since where more time must be spent is in the rest of the modules, as we described before.

The system being designed and developed in [244] will be deployed mostly between the edge and the core of a 5G network, so that early detection of the attack is guaranteed. On the other hand, it will be able to carry out mitigation actions once an attack has been detected, communicating with the Management Framework and making use of SDN techniques to include new security policies in the network such as filtering, redirection to honey-nets, and so on.

On the other hand, we will also make use of Federated Learning techniques to perform the training phase of the proposed AI models, so that we can access more realistic data, and generally private, present only in devices at the edge of the network, increasing the total effectiveness of the whole system. In our proposed federated scheme, several clients that are part of the federated training network, will be able to send and receive data from other 5G clients in order to perform a more effective and less costly training process, and to do so we will guarantee both the anonymity and the privacy of the data shared with SMC and Differential Privacy techniques, among others.

In a separate project effort and due to the heterogeneity of the supported services, an anomaly detection module is being developed, which will be capable of analysing aggregated and fine-grained data such as resource-level data (e.g., computing and storage resource utilization, RAN measurements etc.), flow-level data, service KPIs and infrastructure KPIs as well as traffic patterns and mobility patterns (if available), in order to identify network anomalies and their root cause [238]. This project investigates the anomaly detection problem within the following data analytics types:

- Descriptive analytics to answer "What is happening or happened in network infrastructure?" for detecting the anomaly.
- Diagnostic analytics to answer "Why did it happen or is happening in network infrastructure?" to detect root causes of the anomaly.
- Predictive analytics to answer "What is likely to happen in network infrastructure?" to predict future anomalous behaviour based on previously detected anomalies.
- Prescriptive analytics to answer "What do I need to do in network infrastructure?" to take mitigation action against the root cause of the anomaly to enable fast recovery.

The platform in [238] is expected to manage and orchestrate multiple services simultaneously. For better management of services, the performance of those services should be monitored by means of probes. Those probes are put into user endpoints to continuously monitor the performance status of infrastructure. However, this observation is limited only to the performance of the service and does not contain adequate information regarding the root cause of the problems on the service. In this way, it is impossible to determine the effect of a problem caused by the provider network on the service. Once the problem has been detected by means of probes, mobile network operators can perform root cause analysis by manually operating fault management processes. From the perspective of the service providers or verticals, running each separate service as planned is critical for the correctness and robustness of the provided services over the network infrastructure. The aim of the anomaly detection and root cause analysis is to move towards an automated fault diagnosis tool. This module first identifies the anomalies in the infrastructure and later detects the root cause of the problems. Different states can be used to quantify and represent service quality levels based on observed cumulative KPI values. The states are classified as follows:

- Best: The provided services conditions and all related KPI values are as they should be.
- Good: The provided services conditions are changed with respect to the original best setting due to node failure, link failure or incorrect configuration of routing, and minority of the related KPI values are not within the expected range.
- Fair: The provided services conditions are changed due to faulty nodes, faulty link or bandwidth saturation, and majority of the related KPI values are not within the expected range.
- Bad: There is a faulty network element particularly at the service endpoint in the underlying network topology, and most of the related KPI values are out of the required range.

Transition from one state to another one can take place depending on variations of KPI values.

Anomaly detection and root cause analysis module in the platform aims to identify first the state of the network and later go one more level deep and identify the root cause of being in that state based on the observed KPI values and the help of the recent AI/ML algorithms [238]. To create a training dataset and corresponding model using supervised AI/ML algorithms, a controlled simulator/emulator environment can also be utilized. In this environment, some perturbations on link, node failures, traffic level saturation, congestion, etc. can be created and the effect of each perturbation on relevant KPI values over the whole network can be monitored, labelled and stored to be used for training purposes. After training a supervised model using the training dataset, inferences on the root causes of the problems can be performed using the real-network measurements in the environment. Note that this dataset and model building approach can be beneficial for testing complex topologies that are otherwise hard to be created in real-world environments, e.g. topologies that cover large areas (e.g. a city or country).

4.1.8 Management analytics function

The Management Data Analytics Service (MDAS) [78] follows similar principles to collect, process, analyse and provide useful information to network components as ETSI ENI (c.f., 4.3.2). The MDAS is responsible for all network slices instances, sub-instances and network functions hosted within a network infrastructure. This involves the centralized collection of network data for subsequent publishing to other network management and orchestration modules. In the proposed framework, this service is used to collect mobile data traffic loads generated in the radio access domain by the individual slices. In particular, the MDAS comprises the load level at both NF and network slice levels, provided as a periodic notification and expressed either in absolute terms or relative to the provisioned capacity. As a result, the MDAS allows building historical databases of the network demands for each base station and slice.

In general, MDAS can process and analyse network data detect specific events and even make predictions about network performance. Network data may include performance measurements, traces, Radio Link Failure (RLF) reports, RRC Connection Establishment Failure (RCEF) reports, QoE reports, alarms, configuration data, and various other network analytical data. MDAS provides analytics reports that may include recommended actions [2], that can be enforced at core network level. Finally, MDAS can be exposed, via an Exposure Governance Management Function (EGMF), to external consumers (i.e., other network functions) that may subscribe to customized analytics reports.

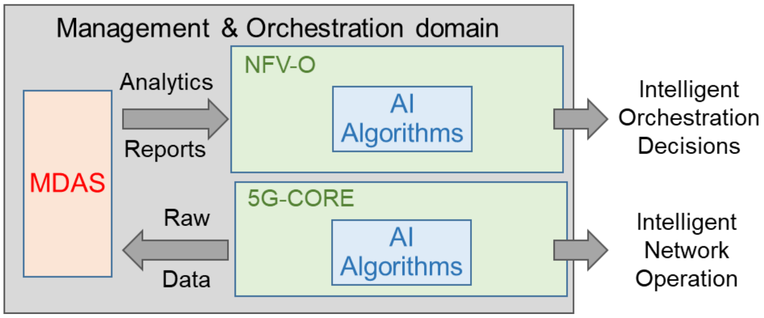

Management & Orchestration domain
MDAS
Analytics
Reports
Raw
Data
NFV-O
AI Algorithms
5G-CORE
AI Algorithms
Intelligent Orchestration Decisions
Intelligent Network Operation

Figure 4-3: The MDAS main functions

4.2 Management of ML models / functions

The need for a coherent approach to management of ML models is recognized by standardization bodies. ITU-T Y.3172 [130] REQ-ML-MNG-004 states that "The ML architecture is required to support an orchestration functionality to manage all the ML functionalities in the network". According to REQ-ML-MNG-004, ML model management includes (i) training, (ii) monitoring, (iii) evaluation, (iv) performance-based selection and (v) reconfiguration/updating of the ML models.

Machine learning is used on many levels of the telecom network stack, as demonstrated by the plethora of examples in this white paper. This begs the questions of management and governance of these models. An AI engine (a term coined in [230]) has been proposed by various projects to provide a central point where ML functions are hosted and managed. This could be one central engine or several isolated or connected engines per network layer.

4.2.1 ML model lifecycle management

ITU-T Y.3172 defines a Machine Learning Function Orchestrator (MLFO) as an entity to support management and orchestration of ML functions in networks. The MLFO is a logical entity that monitors, selects, controls and chains together ML functions based on specified ML Intents or network conditions. The specification defines ML Intent as a declarative description that specifies an ML application. While an MLFO function ideally manages the ML models in an autonomous way, a complete automation of ML model lifecycle management is not yet state of the art.

The following sections discuss the different stages of ML model lifecycle management.

4.2.1.1 ML model training

ITU-T Y.3172 REQ-ML-MNG-004 recognises that there are various ways to perform ML model training (including distributed training), as there are various types of network sources, which generate various types of data. The source of data can be a great concern for ML model training: Is the data available from life networks or can simulated data be used to train the model? How reliable is the collected data, or how accurate is the simulated data? Section 3.2 goes into detail about data considerations, but for the perspective of managing ML models, it is important to know what data a model was trained on to better manage the model lifecycle.

Another aspect to training ML models is where the training takes place. ML models can be trained on premise in the network, on premise close to the data source (e.g. data lake), off-premise in the ML development lab or off-premise in the cloud. Each of the options implies different trade-offs in scalability, security, model accuracy and data access. For example, an ML model that is trained on premise on live network data will be most accurate and up to date but will likely take more time to train due to lower computational resources (compared to cloud) and may cause undesired network disruptions, depending on the type of ML model training[4]. On the other hand, ML models can be trained fast and safely in the cloud, but for that the network data needs to be taken from the network into the cloud. This can cause data protection issues, which in turn can lead to data cleansing/anonymization and therefore a reduction in accuracy for the ML model. The best way to train the ML models has to be determined following a careful use case analysis and prioritisation of the different trade-offs.

4.2.1.2 On-boarding/deployment of ML models

ML models that completed training need to be deployed into the network to start their dedicated tasks of inference, prediction or control. ITU-T Y.3172 REQ-ML-MNG-004 states that ML models should be selected for deployment according to their performance that has been determined in advance. However, depending on how and where the model was trained, issues may arise with regards to unsafe network states in cases where an ML model behaves differently than expected. These differences can be due to inaccurate data or changed network conditions. Models that were trained outside the network and are to be deployed in the network will have to be scrutinised for that reason before they are allowed to interact with a live network. When an ML model fails to provide the desired behaviour, it needs to be reconfigured (e.g. trained on different data). ML deployment management should involve the definition of responsibilities (i) to ensure that the model shows the desired behaviour, and if that is not the case, (ii) to handle the reconfiguration. This is related to trust and accountability for AI/ML (Section 4.4).

4.2.1.3 Monitoring, evaluation and reconfiguration of ML models

A crucial part of ML model lifecycle management is to ensure that a deployed model behaves as desired throughout their lifetime. Even perfect models are likely to degrade eventually due to changes in the environment. This includes loss of accuracy because the data distribution changes (data drift) or the meaning of the data itself changes (concept drift). Therefore, ML models must be continuously monitored. This is also captured by ITU-T Rec. Y.3172 [130], where monitoring includes evaluation of the output of the ML model itself but also of the effect on the network, as measured in network KPIs.

Another aspect to evaluating ML models is defined in ETSI GS ENI 002 OR.1 [97]. The use of AI/ML for intelligent network operation and management shall “minimize the Total Cost of Ownership (TCO), including OPEX (Operating Expenses) and CAPEX (Capital Expenses) of the network infrastructure”. This represents an evaluation of the whole ML approach from a business value perspective, as opposed to the correct working of the technology. The AI/ML hype often ignores the former, with a focus on the latter.

When the performance of an ML model (e.g. measured as prediction error rate or business impact) deteriorates, the model must be reconfigured. This includes updating by replacing a model with a different or more up-to-date one, or removal of an obsolete model. Moreover, [130] requires that ML models should be updated without impact to the network. This also requires continuous monitoring and evaluation, as well as safe transition methods (such as A/B testing or similar techniques) to ensure that the new model does not cause greater harm than the old one.

4.2.2 ML model interface management

In order for deployed ML models to perform their tasks, they need access to data from the network (model input) as well as access to network components that the model is supposed to act on (model output). This

[4] For example, training a reinforcement learning model involves interaction with the environment/network.

is not only another angle to the trust and accountability issue that is relevant for the on-boarding and indeed the whole lifecycle of an ML model, but there is also the point of practicality to achieve efficient and secure ML model I/O. Some AI/ML solutions may include a unified location for data collection, such as a data warehouse or a data lake, which simplifies the data access for ML models. In a similar fashion, a unified access point can provide simplified but also efficient and secure access for the deployed ML models to the network functions.

An AI engine and an intent engine are proposed to address some of the aspect of ML model management [230]. The AI engine is a central host and management platform for the ML models and the Intent engine is a means for platform users, ML models and network to communicate with each other. An example use case for this setup has an ML model in the AI engine analyse network slicing telemetry in order to (re-)configure slice resources through the Slice Manager. The Intent engine serves as a single point of contact to handle the ML model lifecycle (deployment, execution, update, removal), as well as the access to telemetry data, ML function (slice resource prediction) and network function (Slice Manager).

4.3 Standardisation toward enabling AI/ML in networks

4.3.1 3GPP network data analytics function

3GPP has defined NWDAF (Network Data Analytics Function) as part of the Service Based Architecture (SBA) specified in [5]. This logical 5G network function is managed by the 5G operator and it is defined to be capable to collect and process network data from multiple sources and to deliver analytical results to consuming Network or Application Functions responsible for the automated management of different aspects of the 5G network. More specifically, NWDAF collects data from various 5G network functions, such as the Authentication Server Function (AUSF), the Network Exposure Function (NEF), the Unified Data Management (UDM), the NF Repository Function (NRF),the Policy Control Function (PCF), the Session Management Function (SMF), the Access and Mobility management Function (AMF), etc. Via the respective service interfaces and it can provide analytics services [4] to network functions and application functions (AF) as shown in Figure 4-4. NWDAF can also collect information from the Operations Administration and Management (OAM) system, such as NG RAN and 5G core performance measurements and 5G end-to-end KPIs [4].

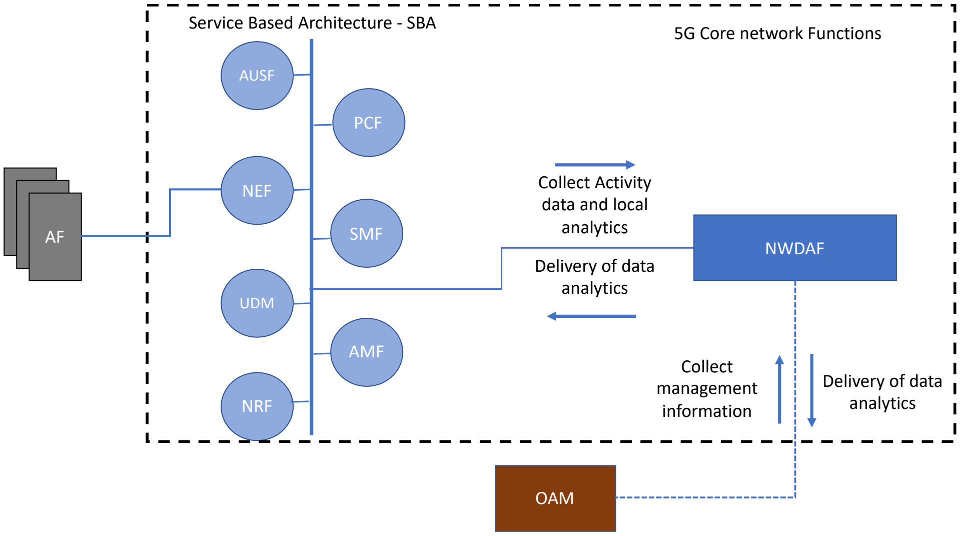

Service Based Architecture - SBA
5G Core network Functions
AUSF
PCF
NEF
SMF
UDM
AMF
NRF
AF
Collect Activity data and local analytics
Delivery of data analytics
NWDAF
Collect management information
Delivery of data analytics
OAM

Figure 4-4 3GPP's Network Data Analytics Function

The analytics services offered by the NWDAF can be either statistical information of the past events, or predictive information, related to a set of potential events briefly summarized in Table 4-1.

Table 4-1 Observed events in NWDAF and potential consumers of analytics

Observed events	Potential consumers
Network Slice load level information	PCF, NSSF
Service experience / QoE	PCF, OAM
Network Function load	AMF (SMF load), SMF (UPF load), OAM
Network performance	PCF, NEF, AF, OAM
Abnormal/Expected UE behaviour	PCF, AMF, SMF, NEF, AF, OAM
UE mobility	AMF, SMF, NEF, UDM, AF, OAM
UE communication pattern	AMF, SMF, NEF, UDM, AF, OAM
User data congestion	NEF, AF
QoS sustainability	PCF, NEF, AF

The NWDAF function can be implemented as a combination of both centralized and distributed elements, in which the different instances may specialize on different types of analytics workloads and handle the entire service more efficiently (c.f., [122]). For example, edge NWDAFs could run real-time analytics and specifically inference tasks at the edges of the 5G network, whilst centralized analytics can cover tasks that are more computationally intensive (e.g., data aggregation, post-processing, Machine Learning etc.)

The internal specification of NWDAF is out of the scope of 3GPP work, neither reference implementations exist or are in progress at SDOs and industrial associations. It can be easily assumed that several of the solutions described in this white paper implement NWDAF functionalities for 5G networks, covering different use cases scenarios and target optimizations. In fact, it is for the 5G operator (through the OAM module) to decide which data to use to populate the information base and models of the NWDAF and how to make use of the data analytics outputs to improve the network performance. Multiple options are possible in that respect. For example:

- The Policy Control Function (PCF) could use analytics inputs to (re-)configure policies for assignment of network resources and/or traffic steering
- The Network Slice Selection Function (NSSF) could optimize Network Slice selection
- The Access and Mobility Management Function (AMF) could improve SMF selection, monitoring of UE behaviour, adjustment of UE mobility
- The Session Management Function (SMF) could improve UPF selection, monitoring of UE behaviour, adjustment of UE communication related network parameters
- The Network Exposure Function (NEF) could optimize forwarding of NWDA information to the AFs
- The Application Function (AF) could adjust service applications
- The OAM could optimize operation and management actions

As mentioned throughout this white paper, the functionality of the NWDAF is of great importance to the optimization of the control and management functions and the overall performance of the network.

Three major aspects are key to the realization of NWDAF solutions:

- *Identification of input data sources*, to be collected at various ends of the 5G network for the different service elements (e.g. network slices, network services, network functions, infrastructure, RAN, etc.) and representative metrics (e.g. logs, %load, position, %availability, %airtime, peak users, etc.). These data require filtering, aggregation and subsequent ingestion into logically centralized data reservoirs where model training can be executed.
- *Selection of the Machine Learning model*, which can best fit the target intelligent optimization target and data characteristics and behaviour (e.g. RL, DRL, RNN, game theory. etc.)
- *Definition of the expected outputs from analytics*, which consist in triggering some imperative (re-)actions on the 5G Network Functions consuming analytics output (e.g. PCF, NSSF, etc.) to implement the planned optimization.

Several solutions in literature point towards such a direction [255], [260]. Through network exposure functionalities, the analytics produced by NWDAF can be consumed by several elements in the network architecture, being one of them the management functions, as described in Section 3.2.3.

4.3.2 ETSI ENI architecture and use case categories

ETSI experiential networked intelligence (ETSI ENI) is an industry standard group (ISG) focusing on improving the operator experience, adding closed-loop AI mechanisms based on context-aware, metadata-driven policies to more quickly recognize and incorporate new and changed knowledge, and hence, make actionable decisions [121]. In viewing of no efficient and extensible standards-based mechanism to provide contextually-aware services (e.g., services that adapt to changes in user needs, business goals, or environmental conditions), ENI launched its activities in 2018, aiming to addressing the following challenges in network operation and management envisaged for 5G networks and beyond. The associated challenges may be stated as [95] automating the manual human operation and determining services to be offered and their associated SLAs, as a function of changing contexts. Such challenges are addressed, from the standard point of view, by defining an architecture framework that provides the mechanisms to observe and learn from operator's experience and to optimize the network operation and management over time. ISG ENI has identified five categories of use cases where AI may benefit network operation and management, summarized in Table 4-2 [95].

Table 4-2 ETSI ENI Use Cases

Category					
1 - Infrastructure Management	Use Case #1-1: Policy-driven IDC Traffic Steering	Use Case #1-2: Handling of Peak Planned Occurrences	Use Case #1-3: DC Energy Saving using AI		
2 - Network Operations	Use Case #2-1: Policy-driven IP Managed Networks	Use Case #2-2: Radio Coverage and Capacity Optimization	Use Case #2-3: Intelligent Software Rollouts	Use Case#2-4: Intelligent Fronthaul Management and Orchestration	Use Case #2-5: Elastic Resource Management and Orchestration
	Use Case #2-6: Application Characteristic based Network	Use Case #2-7: AI enabled network traffic classification	Use Case #2-8: Automatic service and resource design	Use Case #2-9: Intelligent time synchronization of network	

	Operation		framework for cloud service		
3 - Service Orchestration and Management	Use Case #3-1: Context-Aware VoLTE Service Experience Optimization	Use Case #3-2: Intelligent Network Slicing Management	Use Case #3-3: Intelligent Carrier-Managed SD-WAN	Use Case #3-4: Intelligent caching based on prediction of content popularity	
4 - Assurance	Use Case #4-1: Network Fault Identification and Prediction	Use Case #4-2: Assurance of Service Requirements	Use Case #4-3: Network fault root-cause analysis and intelligent recovery		
5 - Network Security	Use Case #5-1: Policy-based network slicing for IoT security	Use Case #5-2: Limiting profit in cyber-attacks			

ENI has defined an architecture [98] that helps address challenges in network automation and optimization using AI. It is expected to enable the assisted system to perform more accurate and efficient decision making. ENI has specified functional blocks and reference points for providing a model-based, policy-driven, context-aware system that provide recommendations and/or commands to assisted systems. This communication may be done directly or indirectly via a designated entity acting on the behalf of the so-called assisted system. A designated entity may be an NMS, an EMS, a controller or in principle any current or future management and orchestration system.

A network has different domains (e.g. RAN/Fixed Access, Transport, and Core). Each domain has its specific functions and services, as well as specific APIs. In a case where the ENI System helps with a localized network function in a specific domain (e.g. optimizing resource allocation at the RAN/Fixed Access), the ENI System may interact with the interfaces of the Assisted Systems of that domain and may collect data from that domain only. In the more likely case, where the ENI System helps with a cross-domain function (e.g. end-to-end network service assurance) the ENI System may interact with multiple domains of the network.

Studies in 5G PPP projects in the area of ML/AI networks are well aligned with the current and future work of ENI. For example in [11], AI based service assurance has been investigated, where the work on AI-based multi-domain service orchestration has been carried out.

4.3.3 O-RAN non-real-time and near-real-time RAN controllers

In relation to O-RAN, the use of the non-real-time RAN intelligent controller (non-RT RIC) is designed to support intelligent RAN optimization by providing policy-based guidance, ML model management and enrichment information to the near-real-time RAN intelligent controller (near-RT RIC) function so that the RAN can optimise, e.g., RRM under certain conditions. It can also perform intelligent radio resource management function in non-real-time interval (i.e., greater than 1 second). The non-RT RIC (i.e., 10 ms to 1 sec.) can use data analytics and AI/ML training/inference to determine the RAN optimization actions for which it can leverage Service Management and Orchestration (SMO) services such as data collection and provisioning services of the O-RAN nodes.

O-RAN, in pursuit of an open and intelligent RAN, has defined AI-powered hierarchical controller structure along with improved open interfaces between decoupled RAN components so that what used to be closed RAN data can be accessed by not only vendors but also operators and 3rd parties to develop innovative RAN applications. In the O-RAN architecture [218], the hierarchically structured intelligent controllers (non-RT RIC and near-RT RIC) enable control and optimization of RAN components and resources using AI/ML models. According to the specification of O-RAN, RIC supports the entire AI/ML workflow, including measurement data collection, data processing, and training/inference/update of the AI/ML models.

The non-RT RIC embedded in the SMO layer is the intelligent management centre that fulfils non-real-time control. O-RAN specification defines an interface (i.e., A1) to connect non-RT RIC in the SMO with the near-RT RIC elements. The RAN data can then be acquired and consumed by the non-RT RIC via the A1 Interface. The non-RT RIC also supports information enrichment that may come from external data sources or be extracted with AI approaches from the historical RAN data. The AI/ML models learned in the non-RT RIC can be used by the SMO to analyse RAN and generate optimization operations for improvement of the E2E user service experience and the network performance. In addition, the non-RT RIC also provides model-training for the near-RT RIC [219]. The non-RT RIC is closely related to the MDAF module mentioned earlier (c.f., Section 4.3.1) as the key function is to enable intelligent network management. Thus, it could be viewed as an MDAF instance with management data analytic service.

The near-RT RIC located in the radio side enables fine-grained data collection and near real time RAN control over the E2 interface. The near-RT RIC provides embedded intelligence, as well as a platform for on-boarding of third-party control-applications. The near-RT RIC can leverage data about the near real-time state of the underlying network via the Radio-Network Information Base (R-NIB). Since training of AI/ML models could be time consuming due to complexity and size of the data, it is often moved to the non-RT RIC, and the learned models are conveyed to the near-RT RIC via the A1 interface. The near-RT RIC is thus able to perform inference efficiently to achieve network analytics and optimization with a tighter timing requirement.

4.4 Trust in AI/ML-based networks

4.4.1 Privacy concerns

Privacy concerns are a traditional topic that is under continuous research due to the emergence of new telecommunication networks, technologies, services and methods. To address these concerns, ML and DL techniques have been a recurrent theme in recent years. 5G and beyond 5G networks are considered a novel environment where offered services will handle paramount information about users. Many of these services will utilize ML/DL algorithms to process and analyse user’s data in order to provide more suitable services. In this sense, recent studies demonstrated that ML/DL techniques can reveal private personal information and compromise user’s privacy [153], therefore, service providers must clarify to user how their data will be used, where they will be stored, and what purpose of their use is.

As aforementioned, ML/DL techniques are continuously handling user’s data, and consequently, another concern about the processing of data by ML/DL techniques appears in the technique of data anonymization. Some techniques of anonymizing data utilize private details elimination or replacement them with random values, which is inadequate and it can compromise data privacy since that information could be recovered by an adversary obtaining auxiliary information about the individuals represented in the dataset [110]. These anonymizing techniques are classified as non-cryptographic solutions and they contain differential privacy approaches [268]. In contrast, ML/DL can also incorporate cryptographic means such as [227] multiparty computation, homomorphic encryption, and zero-knowledge argument schemes, which enable models to learn while preserving the privacy of information. The most well-known distributed approach is Federated Learning [189], where learning is distributed to multiple places in order to enhance efficiency, security, and privacy.

Over and above previous attacks, other conventional attacks again privacy in machine learning and deep learning approaches are property inference attacks and model inversion and attribute inference attacks. First and foremost, property inference attacks intend to deduce patterns of information from the objective model such as the memorization attacks which attempt to detect sensible sequences from the training data of the objective model [64]. These attacks are broadly carried out on neural networks [109] and hidden Markov models [45], and they could be mitigated using differential privacy techniques (such as gradient perturbation) or a secure multi-party computation. Secondly, model inversion and attribute inference attacks are focused on inferring sensitive attributes of data instance from a publicly-released model [291]. In order to cover these attacks, techniques such as information-theoretic privacy or homomorphic encryption could be utilized. In the end, there is a set of prevalent privacy-preserving mechanisms (see Figure 4-5) which can be applied depending on the phase where it is necessary to provide that extra level of privacy.

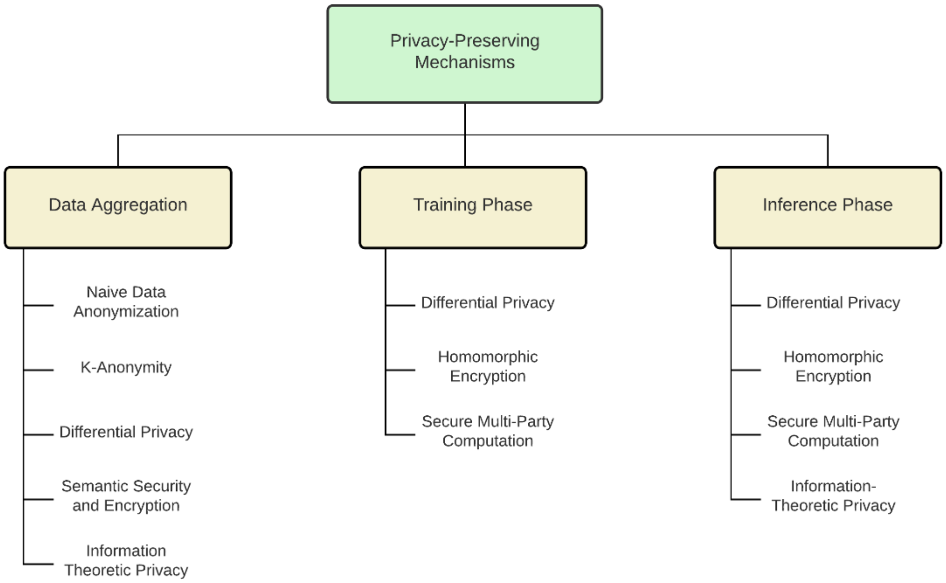


Figure 4-5 Categorization of privacy-preserving schemes for ML and DL [196].

4.4.2 Trustworthy AI/ML

A general topic that is currently not solidly addressed is how we can construct trustworthy AI/ML-based systems before they are incorporated in future networks. Towards this direction three research avenues are important to consider: (i) resilient machine learning, (ii) explainable artificial intelligence and (iii) human in the loop (HITL).

Resilient ML – ML algorithms and in particular deep neural networks (DNNs) have been found vulnerable to adversarial attacks, malicious and well-designed examples than can fool a model with little perturbations added to the input and that cannot be detected by human experts [204]. Therefore, the adversarial attacks could mislead the ML-based systems and cause harmful situations in security-critical areas of the network [201]. When protecting a ML model against adversarial attacks, a recently introduced "no free lunch" theorem establishes a trade-off between accuracy and robustness [226]. Although perfectly effective defences against adversarial traffic are yet to be proposed, one of the most successful verification processes to date is adversarial training [129]. Two recently released open source libraries called Cleverhans [203] and Foolbox [135] allow testing ML components against attacks by means of adversarial training. These libraries contain reference implementations of several attack and defence procedures to be used to test ML models against standardized, state-of-the-art attacks and defences. Further work is needed to develop new resilient ML-based components enhancing the Cleverhans and Foolbox libraries for designing robust ML and DL models that can defend themselves against malicious adversarial examples. Possible defence techniques include input transformation, gradient masking and denoising, and verification ensembles against attacks such as adversarial perturbations, out of distribution black box attacks and white box attacks

Explainable AI – Explainability of AI is critical in a wide range of mission-critical services and safety-critical task management. Due to the complex nature of DNNs, it is difficult to interpret AI outputs for a human operator when consulted to oversee a decision. Performance versus explainability trade-offs appear and therefore, for critical operations, easily interpretable simpler linear models or decision trees are often preferred over complex models that provide higher accuracy. There is growing research in the field of explainability of deep networks in order to safely introduce them into critical operations [114] and thus several models of explainability have been proposed. Reducing data and DNNs models by applying the recently appeared Automatic ML techniques [295] is also a research trend that is gaining momentum. Another approach is to extract information from learned models and present the underlying reasoning to a human operator in an easy to understand format using natural language interpretations, visualizations of learned representations or models, and comparison to similar cases. At the moment, a wide range of machine and deep learning models contributing to different network OAM tasks are not well understood. The need for increased explainability to enable trust is crucial for the applicability of AI in network OAM.

Human in the Loop (HITL) – Accuracy assessment of AI predictions is critical to evaluate whether an AI module is working as intended by its design. Modern AI algorithms provide prediction accuracies by infusing uncertainty into the predictions and evaluation metrics. A prominent approach is to use Bayesian Deep Learning (BDL) algorithms [108], which outputs distributions rather than point estimates to provide uncertainty estimations (confidence intervals, credible intervals). On the other hand, an alternative approach is to use Bayesian Neural Networks (BNN) as a decoupled stage to map uncalibrated, raw DNN outputs into ones that can be reliably interpreted as probabilities. Once the accuracy is determined, critical decision making can be handled through human-assisted mechanisms, in areas related to lack of previous data to provide sufficient training, as well as in the cases of rare events that require some form of human intervention. Traditional approaches mainly consider HITL mechanisms during the training phase in the absence of labelled data, but without intention of avoiding actions that can be detrimental to the application performance [251]. Further research is needed on the issues of preventing detrimental actions, while quality of the decision making is also improved alongside [75].

4.4.3 Zero trust management

Trust is a subjective and abstract concept that is commonly familiarized with computer science area for ages. Nevertheless, the emergence of 5G telecommunication network along with the growth of interconnected devices has created vulnerable fronts against various security threads. Zero-trust is a security concept centred on the idea that entities should not automatically trust anything inside or outside their perimeters, instead, they must verify anything and everything trying to connect to their systems before granting service access.

In 2010, the analyst firm, Forrester Research [154], presented the first zero-trust network management approach. The fundamental pillar of this approach was simple and straightforward, “never trust, always verify”. Nonetheless, the zero-trust concept has adapted to the requirements and changes in networks, thence the concept currently covers paramount principles such as entities and devices access control, continuous network access control, data control, visibility and analysis, workloads, and so on [166]. Recently, National Institute of Standards and Technology (NIST) has published a proposal of zero-trust architecture which considers the most essential requirements of 5G networks [254]. According to NIST’s recommendations, zero-trust architecture should be primarily focused on data and service protection, but it may also be extended to assets and subjects. In addition, a zero-trust approach should involve continuous analysing and evaluating of the risks of the assets, as well as presenting protection to mitigate those risks. Such architecture encompasses fundamentals such as identity, credentials, access management, operations, endpoints, hosting environments, and infrastructure interconnections. Therefore, this architecture could be summarized as an end-to-end approach to ensure the security of data and resources.

Through zero-trust network management, it is possible to decrease security risk and attack surface at inter/ intra-domain level, even though no trust relationship is ever granted implicitly. Thence, a zero-trust approach introduces fewer vulnerabilities and threats than conventional network management models. For instance, it will be feasible to identify and reduce the attacker's ability to carry out network lateral movement attack. Furthermore, this approach also provides greater data protection since a smart data segmentation is necessary to regulate access control. Another essential characteristic is the capacity to dynamically support access control based on the current use cases. A real scenario could be when a stakeholder detects lack of capability to address the tasks agreed through an SLA, and therefore, it needs to request resources from a third-party resource provider. In this case, a set of resources/services will be shared across multiple administrative domains, and consequently, it will be essential to continuously assess the trust level, establish access control policies, and perform identity management of the entities that will use the resources, in order to generate an end-to-end zero trust relationship.

As a consequence, the introduction of zero trust management involves a set of requirements and features that must be considered in order to guarantee security and trustworthiness. First and foremost, identity management is a mandatory requirement in zero-trust approaches due to the fact that assets, subjects, services and resources should be recognized before granting access. Second, access control since such model should consider the implementation of the least-privilege as a basis. Last but not least, the model should be able to learn and adapt to current situations, hence zero-touch management is another key aspect to be introduced in zero-trust management.

4.4.4 Widely available data-sets

The success of AI/ML models in a variety of network applications and services relies heavily on the use of network data in diverse levels of granularity. Publicly available real and simulated benchmark datasets play an important role in model development and evaluation, as well as fair comparison with state-of-the-art solutions.

As discussed in previous sections the training of AI/ML algorithms is requiring large amounts of data, which are typically not readily available for many reasons. In research projects often the amount of data that can be generated with the prototype systems and experimentation use cases do not suffice to efficiently train the algorithms. Access to readily available data is needed; however so far no sustainable initiatives have emerged that attempted to create a large repository of network traffic data from the different network domains. Importantly, data may contain privacy related information that needs to be sustainably anonymised before it can be provided to an open repository. Similarly, data, especially from commercial networks, typically contain business related information that need to be cleansed as well.

Due to concerns of data privacy of end users and commercial confidentiality of network operators and service providers, it is difficult for third parties to obtain network data. Typical accesses to public datasets include:

- Open-access data/code repositories maintained e.g. on Zenodo, GitHub, and Kaggle.
- International competitions organized by network related conferences and organizations together with operators and service providers.
- Projects funded by the EU Framework programs (H2020/ 5G PPP) [14], [13].
- Scientific publications providing datasets used to evaluate the proposed AI-driven solutions. Due to requirements of replication and reproduction, an increasing number of papers published in conferences and journals intend to release the datasets to the research community.

In the context of European projects, telecom operators have proposed the idea of building a repository of open data sets, however with very limited success so far. The idea of choosing a pre-competitive environment such a European framework programmes to build an open repository, seems attractive and could be the best environment to overcome the potential concerns. Considering the heavy dependency of AI/ML on such open training data, it is worthwhile to consider launching an open initiative for creating such a repository in the near future.

Alternatively, open source software (OSS) is often used to produce simulated datasets for development of AI-driven network applications. For example, O-RAN Bronze [220] and FlexRAN [105] are used for the RAN domain, OpenAirInterface [214] for EPC/RAN/E-UTRAN, OSM [213] for NFV MANO, as well as OpenStack [215] and Kubernetes [156] for NFVI. Importantly, all the public datasets and OSS should be leveraged under the open source licenses required by the owners/authors.

4.4.5 Stochastic geometry for network analysis and optimization

In the context of generation of large data sets, stochastic geometry is a powerful tool for modelling, analysing, and optimizing large-scale and ultra-dense cellular networks [278]. A major research problem in current and future wireless networks is to optimize the deployment density of a cellular network, given the transmit power of the base stations and aiming at energy-efficiency optimization. Assuming that the base stations are distributed according to a Poisson point process, an accurate and realistic analytical model for optimizing energy efficiency was recently proposed in [278] and formulated the optimal deployment density of the cellular base stations in a tractable analytical form. By leveraging the approach described in [278], large data sets can be generated with low computational effort. This provides system designers with the optimal base station density as a function of the base station transmit power. However, similarly tractable analytical frameworks cannot be easily obtained if cellular base stations are distributed according to non-Poisson spatial models, so energy-efficiency optimization of such cellular network deployments is difficult. In addition, the generation of large data sets based on non-Poisson point processes is a time- and memory-consuming task, making it hard to obtain large data sets containing optimal data pairs (optimal deployment density and transmit power for training purposes). This is a typical example in which a tractable model is available (based on the Poisson point process), but it is not sufficiently accurate. Nevertheless, even inaccurate models can provide system designers with useful information that should not be missed. In general, employing a fully data-driven approach to train an artificial neural network (ANN) requires the acquisition of a huge amount of live data. This task, however, might not be practical due to time, complexity, or economic factors in the process of acquiring data. Instead, the availability of an approximate model can be exploited to perform a first rough training of the

ANN, which can be subsequently refined through a small set of real data. This approach is likely to reduce the amount of live data needed for training ANNs. Therefore, in this context, an approach that combines approximate modelling based on stochastic geometry and data driven methods is well motivated. In [266] and [246], for example, the authors assume that the base stations are distributed according to a non-Poisson point process, whose exact distribution is not known, and that only (some) empirical samples for the locations of the base stations are available. The authors aim to understand whether by first performing an initial training of the ANN based on a large Poisson-based data set and then executing a second training based on a small data set of empirical (or synthetic from simulations) data, we can obtain a performance similar to that obtained using only a large training set of real data. The results obtained in [266] and [246] clearly show that pre-training an ANN by using a Poisson-based data set and then refining it by using few non-Poisson data is possible and lead to the same results as training an ANN by using large data sets of non-Poisson data.

4.5 AI/ML-based KPI validation and system troubleshooting

The instantiation of a Monitoring and Analytics (M&A) framework is key in modern communication systems, and 5G exacerbates this requirement [91]. In particular, 5G services have to comply with SLAs, which state the E2E KPIs that have to be guaranteed to end-users and verticals. This leads to the need for automated monitoring and management of the instantiated resources, in order to promptly identify network bottlenecks and system malfunctions that hinder the compliance with SLAs. A reliable and efficient M&A framework should thus consider both end-users’ and operators’ perspectives, aiming at improving systems’ QoS and users’ QoE, while minimizing operators’ management and operational costs.

Within the above context, a 5G PPP infrastructure project [234] targets the realization of a full-chain M&A framework, for a reliable validation of 5G KPIs [236]. The framework enables the analysis of experimental data collected by dedicated monitoring probes during the usage of the experimentation facility. This in turn allows for pinpointing the interdependencies between network configurations, scenario conditions, and QoS/QoE KPIs, ultimately leading to the derivation of optimized management policies for further improvement of users’ and verticals’ performance.

The M&A framework includes several monitoring tools and both statistical and ML-based analytics. It is formed by three main blocks:

- *Infrastructure Monitoring*, which focuses on the collection of data on the status of infrastructure components, e.g., user equipment, radio access and core networks, SDN/NFV environments, and computing and storage distributed units;
- *Performance Monitoring*, which is devoted to the active measure of E2E QoS/QoE KPIs. These include traditional indicators, such as throughput and latency, but also other indicators tailored on specific use cases and applications (e.g., for mission critical services);
- *Storage and Analytics*, which enables efficient management of large amounts of heterogeneous data, and drives the discovery of hidden values, correlation, and causalities among them.

Among others, the M&A framework aims at providing the following analytics functionalities:

KPI validation, i.e., the execution of the KPI statistical analysis defined in [237].

1. *Time series management*, which allows to coherently merge the data coming from different probes, in order to perform further analyses. In a M&A system, this task is needed for several reasons. First, different sampling rates might be used by different probes. For example, QoS/QoE KPIs might be collected at higher sampling rates compared to infrastructure data. Second, the probes might be not perfectly synchronized. Hence, synchronization is applied if the time series collected from different probes present similar sampling rates, while interpolation better suits situations where the probes use different sampling rates.
2. *Outlier detection*, in order to eliminate data obtained under incorrect functioning of the probes, which may negatively affect the analyses. Classical approaches include Z-score and Modified Z-score, which consider data sample statistics. Other approaches exploit ML algorithms, such as SVM, Isolation Forest, and Autoencoders, among others.
3. *Feature selection*, which allows simplifying the analyses by eliminating some of the collected parameters. As a matter of fact, 5G networks include a huge number of components. Hence, using AI/ML approaches for network management and optimization could be challenged by the large amount of data that can be potentially collected; some of these data might also be not useful and could negatively affect the analyses. Hence, feature selection algorithms (e.g., Recursive Feature Elimination, Backward Elimination, and Least Absolute Shrinkage and Selection Operator – LASSO – regression, among others) can be used to remove redundant features, making the next analyses computationally simpler and faster. In general, feature selection allows training ML algorithms faster, reducing model complexity and overfitting, and improving model accuracy.
4. *Correlation analysis*, which allows highlighting how system configurations and network conditions, collected via IM probes, are correlated and affect QoS/QoE KPIs, collected via performance monitoring tools. Revealing the correlation between infrastructure and performance monitoring parameters allows improving network management and deriving better configuration policies for assuring SLAs. Lack of correlation between parameters which are known to have dependencies is also a key indicator pinpointing system malfunctioning and trigger needed reactions.
5. *KPI prediction*, which allows building a model and estimating QoS/QoE KPIs by looking at other parameters, collected under different circumstances and scenarios. Several supervised ML algorithms can be used, such as Linear and Support Vector Regression and Random Forest, among others. Being able to accurately predict a KPI would enable better network planning and management.
6. *KPI time series forecasting*, in order to build a model over time for QoS/QoE KPIs, thus deriving nominal trends and forecasting next-future patterns. Also in this case, several algorithms can be used, including Seasonal ARIMA and LSTM networks. The deviation from the expected pattern can be used as an indication of anomalous behaviors due to unexpected reasons, and may trigger a network alarm system for prompt reaction aiming at restoring nominal trends.

Summary and recommendations

This white paper on AI/ML as enablers of 5G and B5G networks is based on contributions from 5G PPP projects that research, implement and validate 5G and B5G network systems.

The white paper introduces the main relevant mechanisms in Artificial Intelligence and Machine Learning currently investigated and exploited for 5G and beyond 5G networks. A family of neural networks is presented, which are generally speaking, non-linear statistical data modelling and decision making tools. They are typically used to model complex relationships between input and output parameters of a system or to find patterns in data. Feed-forward neural networks, deep neural networks, recurrent neural networks, and convolutional neural networks belong to this family. Reinforcement learning is concerned about how intelligent agents must take actions in order to maximize a collective reward, e.g. to improve a property of the system. Deep reinforcement learning combines deep neural networks and has the benefit that is can operate on non-structured data. Hybrid solutions are presented such as combined analytical and machine learning modelling as well as expert knowledge aided machine learning. Finally other specific methods are presented, such as generative adversarial networks and unsupervised learning and clustering.

In the sequel the white paper elaborates on use case and optimisation problems that are being tackled with AI/ML, partitioned in three major areas, namely: network planning, network diagnostics/insights, and network optimisation and control. In network planning, attention is given to the network element placement problem and to dimensioning considerations for C-RAN clusters. In network diagnostics, attention is given to forecasting network conditions, characteristics and undesired events, such as security incidents. Estimating user location is part of network insights. Finally, in network optimisation and control attention is given to the different network segments, including RAN, transport networks, fronthaul and backhaul, virtualisation infrastructure, end-to-end network slicing, security and application functions.

The white paper discusses the application of AI/ML in the 5G network architecture. In this context is identifies solutions pertaining to AI-based autonomous slice management, control and orchestration, AI/ML-based scaling operations in network service orchestration, AI/ML as a Service in network management and orchestration, enablement of ML for the verticals' domain, cross-layer optimization, management analytics in general, 3rd party ML analytics for network operation optimization in particular, anomaly detection using AI/ML. In the context of architecture it discusses the requirements for ML model lifecycle and interface management. Furthermore it investigates the global efforts for the enablement of AI/ML in networks, including the network data analytics function, the lack of availability of data-sets for training the AI/ML models and the associated privacy concerns. Finally, it identifies the challenges in view of trust in AI/ML-based networks and potential solutions such as the zero-trust management approach. The section concludes with a brief overview of AI/ML-based KPI validation and system troubleshooting.

In summary the findings of this white paper conclude that for enhancing future network return on investment the following areas need further attention (research and development work):

(g) building standardized interfaces to access relevant and actionable data,
(h) exploring ways of using AI to optimize customer experience,
(i) running early trials with new customer segments to identify AI opportunities,
(j) examining use of AI and automation for network operations, including planning and optimization,
(k) ensuring early adoption of new solutions for AI and automation to facilitate introduction of new use cases, and
(l) establish/launch an open repository for network data sets that can be used for training and benchmarking algorithms by all

Abbreviations

2D	Two-Dimensional
3D	Three-Dimensional
3GPP	3rd Generation Partnership Project
5G PPP	5G Public Private Partnership
5GC	5G Core
AaAaaS	Acronyms and Abbreviations as a Service
AAoA	Azimuth Angle of Arrival
AAoD	Azimuth Angle of Departure
ABR	Adaptive Bitrate
ACL	Autonomic Control Loop
AGV	Automated Guided Vehicle
AI	Artificial Intelligence
AIMLaaS	AIML as a Service
AIMLP	AIML Platform
AMF	Access and Mobility management Function
ANM	Autonomic Network Management
ANMO-SDR	Autonomic Network Management Optimisation SDR
ANN	Artificial neural networks
AoA	Angle of Arrival
AoD	Angle of Departure
AP	Access Point
APEX	Adaptive Policy Execution
API	Application Programming Interface
AQM	Active Queue Management
ARIMA	Auto Regressive Integrated Moving Average
ASIC	Application-Specific Integrated Circuit
ATSSS / AT3S	Access Traffic Steering, Switching & Splitting
AUSF	Authentication Server Function
B5G	Beyond 5G
BBU	Baseband Unit
BGP	Border Gateway Protocol
BH	Backhaul

BP	Backward Propagation
BPNN	Back-Propagation Neural network
BWMS	Bandwidth Management Service
CAPEX	Capital Expenditure
CDN	Content Delivery Network
CER	Crossover Error Rate
CI/CD	Continuous Integration / Continuous Delivery or Deployment
CIR	Channel Impulse Response
CN	Core Network
CNF	Cloud native Network Function
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CQI	Channel Quality Information
C-RAN	Cloud Radio Access Network
CRN	Cognitive Radio Network
CU	Central Unit
CUDA	Compute Unified Device Architecture
D2D	Device to Device
DASH	Dynamic Adaptive Streaming over HTTP
DBN	Deep Belief Network
DDoS	Distributed Denial of Service
DDPG	Deep Deterministic Policy Gradient
DDQN	Double Q-Learning
DL	Downlink
DNN	Deep Neural Network
DOI	Digital Object Identifier
DPG	Deterministic Policy Gradient
DPP	Data Plane Programmability
DQN	Deep Q-Learning Network
DRL	Deep Reinforcement Learning
DU	Distributed Unit
E2E	End-to-End
EAoA	Elevation Angle of Arrival
EAoD	Elevation Angle of Departure

eAT3S	enhanced AT3S
ECDF	Empirical Cumulative Distribution Function
E-CID	Enhanced-Cell ID
ECN	Explicit Congestion Notification
EGMF	Exposure Governance Management Function
EKF	Extended Kalman Filter
eMBB	Enhanced Mobile Broadband
EMS	Element Management System
eNB	eNodeB, Base Station in 4G/LTE
ENI	Experiential Networked Intelligence (ETSI)
EOP	Edge Orchestration Platform
EPC	Evolved Packet Core
ESN	Echo State Network
ETS	Error, Trend, Seasonality
ETSI	European Telecommunications Standards Institute
EU	European Union
E-UTRAN	Evolved UMTS Terrestrial Radio Access Network
FCAPS	Fault, Configuration, Accounting, Performance, Security
FDMA	Frequency-Division Multiple Access
FF	Feed Forward
FFNN	Feed Forward Neural Network
FH	Fronthaul
FL	Federated Learning
FPGA	Field Programmable Gate Array
FQL	Fuzzy Q-Learning
GA	Genetic Algorithm
GAN	Generative Adversarial Network
gNB	gNodeB, Base Station in 5G
GPS	Global Positioning System
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
GS	Group Specification (ETSI)
GSMA	GSM Association
GST	Generic Network Service Template (GSMA)

HSS	Home Subscriber Server
ICC	International Conference on Communications
ICT	Information and Communication Technologies
IEEE	Institute of Electrical and Electronics Engineers
ILP	Integer Linear Programming
IOO	Indoor Open Office
IoRL	Internet of Radio Light
IP	Internet Protocol
ISG	Industry Specification Group (ETSI)
ITU-T	International Telecommunication Union - Telecommunication Standardization Sector
KF	Kalman Filter
k-NN	k-Nearest Neighbours
KPI	Key Performance Indicator
KQI	Key Quality Indicator
LAN	Local Area Network
LASSO	Least Absolute Shrinkage and Selection Operator
LED	Light-Emitting-Diode
LiFi	Wireless communication technology over visible light
LoS	Line of Sight
LS-SVM	Least Square Support Vector Machine
LSTM	Long Short Term Memory
LTE	Long Term Evolution
M&A	Monitoring and Analytics
M2M	Machine to Machine
MAC	Medium Access Control
MAD	Median Absolute Deviation
MANO	Management and Orchestration
MAPE	Monitor-Analyse-Plan-Execute
MAPE-K	Monitor-Analyse-Plan-Execute-Knowledge
MARL	Multi-Agent Reinforcement Learning
MDAS	Management Data Analytics Service
MDP	Markov Decision Process
MEC	Multi-access Edge Computing
MEO	Mobile Edge Orchestrator

MIMO	Multiple-Input and Multiple-Output
mIoT	Massive Internet of Things
ML	Machine Learning
MLFO	Machine Learning Function Orchestrator
MLP	Multilayer Perceptron
MME	Mobility Management Entity
mMTC	Massive Machine Type Communications
MN	Mobile Node
MNO	Mobile Network Operator
MOCN	Multi Operator core Network
MPTCP	Multi-Path TCP
MSE	Mean Square Error
MTD	Moving Target Defence
NEF	Network Exposure Function
NF	Network Function
NFVI	Network Function Virtualisation Infrastructures
NIST	National Institute of Standards and Technology
NLoS	Non-Line-of-Sight
NLP	Non-Linear Programming
NMS	Network Management System
NN	Neural Network
NP	Nondeterministic Polynomial time
NR	New Radio
NRF	NF Repository Function
NSD	Network Service Descriptor
NSP	Network Service Provider
NSSF	Network Slice Selection function
NT	Network Tomography
NWDAF	Network Data Analytics Function
OAM	Operations Administration and Management
ONAP	Open Network Automation Platform
OPEX	Operational Expenditure
OSA	One Stop API
OSM	Open source MANO

OSS	Open Source Software
OWC	Optical wireless Communication
PCF	Policy Control Function
PHY	Physical layer
PnP	Perspective-n-Point
PoP	Point-of-Presence
PPO	Proximal Policy Optimization
PPP	Poisson Point Process
PRB	Physical Resource Block
PRS	Positioning Reference Signal
PSO	Particle Swarm Optimisation
PUSCH	Physical Uplink Shared Channel
QoE	Quality of Experience
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RAN	Radio Access Network
RAT	Radio Access Technology
RCA	Resource Channel Allocation
ReLU	Rectified Linear Unit
REST	Representational State Transfer
RF	Radio Frequency
RIC	Radio Intelligent Controller
RIS	Reconfigurable Intelligent Surface
RL	Reinforcement Learning
RL	Resource Layer
RMS	Root Mean Square
RMSE	Root Mean Square Error
RNIS	Radio Network Information Service
RNN	Recurrent Neural Network
RRM	Radio Resource Management
RSI	RAN Slice Instance
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RSS	Received Signal Strength

RSSI	Received Signal Strength Indication
RT	Real-Time
RTT	Round Trip Time
RU	Remote Unit
RVM	Relevance Vector Machine
SA	Service Assurance
SARSA	State–action–reward–state–action
SBA	Service based Architecture
SDN	Software Defined Networking
SDO	Standards Developing Organisation
SDR	Software Defined Radio
SFC	Service Function Chaining
SI	Soft Information
SINR	Signal-to-Interference-plus-Noise Ratio
SLA	Service Level Agreement
SMF	Session Management Function
SMO	Service Management and Orchestration
SNPN	Standalone Non-Public Network
SNR	Signal to Noise Ratio
SO	Service Orchestrator
SOM	Self-Organising Map
SP	Service provider
SPGW	Serving Gateway/Packet Data Network Gateway
SSID	Service Set Identifier
SVE	Single Value Estimation
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TD3	Twin Delayed DDPG
TDOA	Time Difference of Arrival
TMF	Tele-Management Forum
UDM	Unified Data Management
UE	User Equipment
UKF	Unscented Kalman Filter
UL	Uplink

UPF	User Plane Function
URLLC	Ultra-Reliable Low Latency Communications
UWB	Ultra-Wideband
V2N	Vehicle to Network
VIM	Virtualised Infrastructure manager
VLC	Visual Light Communication
VM	Virtual Machine
VNF	Virtualised Network Function
VNI	Virtual Network Indicator
VoMS	Vertical oriented Monitoring Service
vRAN	Virtualised RAN
VS	Vertical Slicer
VUF	Virtual Utility Function
VxLAN	Virtually extensible Local Area Network
WDM	Wavelength Division Multiplexing
WLAN	Wireless Local Area Network
ZSM	Zero Touch Network and Service Management (ETSI)

References

[1] 3GPP TR 28.800: Technical Specification Group Services and System Aspects; Telecommunication management; Study on management and orchestration architecture of next generation networks and services

[2] 3GPP TR 28.809: Study on enhancement of management data analytics

[3] 3GPP TR 38.855: *Study on NR positioning support (Release 16)*, V16.0.0, March 2019, accessed 22 January 2021. https://www.3gpp.org/ftp//Specs/archive/38_series/38.855/

[4] 3GPP TS 23.288: Architecture enhancements for 5G System (5GS) to support network data analytics services.

[5] 3GPP TS 23.501, system Architecture for the 5G System (5GS), Stage 2, Release 16, Sept. 2020

[6] 3GPP TS 28.530: Technical Specification Group Services and System Aspects; Management and orchestration; Concepts, use cases and requirements

[7] 3GPP TS 28.533: Management and orchestration; Architecture framework

[8] 3GPP TS 29.520, Network Data Analytics Services, Stage 3, Release 17, Sept. 2020

[9] 5G IA, June 2020, European Partnership under Horizon Europe Smart Networks and Services, https://ec.europa.eu/info/sites/info/files/research_and_innovation/funding/documents/ec_rtd_he-partnership_smart-networks-services.pdf

[10] 5G PPP, 5G and Verticals, online: https://5g-ppp.eu/verticals/

[11] 5G Vertical Innovation Infrastructure, Project 5G-VINNI, online: https://www.5g-vinni.eu/

[12] 5G Vision. The 5G Infrastructure Public Private Partnership: the next generation of communication networks and services. 2015. [Online]

[13] 5GPPP Phase 2 Projects. https://5g-ppp.eu/5g-ppp-phase-2-projects/

[14] 5GPPP Phase 3 Projects. https://5g-ppp.eu/5g-ppp-phase-3-projects/

[15] A. A. Boulogeorgos and A. Alexiou, "Error Analysis of Mixed THz-RF Wireless Systems," in IEEE Communications Letters, vol. 24, no. 2, pp. 277-281, Feb. 2020, doi: 10.1109/LCOMM.2019.2959337.

[16] A. A. Boulogeorgos and A. Alexiou, "Outage Probability Analysis of THz Relaying Systems," 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, London, UK, 2020, pp. 1-7, doi: 10.1109/PIMRC48278.2020.9217121.

[17] A. A. Boulogeorgos and A. Alexiou, "Performance evaluation of the initial access procedure in wireless THz systems," 2019 16th International Symposium on Wireless Communication Systems (ISWCS), Oulu, Finland, 2019, pp. 422-426, doi: 10.1109/ISWCS.2019.8877185.

[18] A. A. Boulogeorgos et al., "Terahertz Technologies to Deliver Optical Network Quality of Experience in Wireless Systems Beyond 5G," in IEEE Communications Magazine, vol. 56, no. 6, pp. 144-151, June 2018, doi: 10.1109/MCOM.2018.1700890.

[19] A. A. Boulogeorgos, E. N. Papasotiriou and A. Alexiou, "Analytical Performance Assessment of THz Wireless Systems," in IEEE Access, vol. 7, pp. 11436-11453, 2019, doi: 10.1109/ACCESS.2019.2892198.

[20] A. A. Boulogeorgos, S. K. Goudos and A. Alexiou, "Users Association in Ultra Dense THz Networks," 2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, 2018, pp. 1-5, doi: 10.1109/SPAWC.2018.8445950.

[21] A. Carron, M. Todescato, R. Carli, L. Schenato, and G. Pillonetto, “Machine learning meets Kalman Filtering,” 2016 IEEE 55th Conf. Decis. Control. CDC 2016, vol. 1, no. Cdc, pp. 4594–4599, 2016, doi: 10.1109/CDC.2016.7798968

[22] A. Conti, S. Mazuelas, S. Bartoletti, W. C. Lindsey and M. Z. Win, "Soft Information for Localization-of-Things," in Proceedings of the IEEE, vol. 107, no. 11, pp. 2240-2264, Nov. 2019, doi: 10.1109/JPROC.2019.2905854

[23] A. Giannopoulos, S. Spantideas, N. Kapsalis, P. Gkonis, P. Karkazis, L. Sarakis, P. Trakadas and C. Capsalis, Demand-Driven Power Allocation in 5G-enabled Wireless Networks with Deep Q-Learning. 22nd IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM) (to be held in Pisa, 2021, under review

[24] A. Gupta et al., |On service-chaining strategies using Virtual Network Functions in operator networks”, *Computer Networks*, Vo. 133, 14 March 2018,

[25] A. J. Muhammed, Z. Ma, Z. Zhang, P. Fan and E. G. Larsson, "Energy-Efficient Resource Allocation for NOMA Based Small Cell Networks With Wireless Backhauls," in IEEE Transactions on Communications, vol. 68, no. 6, pp. 3766-3781, June 2020, doi: 10.1109/TCOMM.2020.2979971.

[26] A. Laghrissi and T. Taleb, "A Survey on the Placement of Virtual Resources and Virtual Network Functions," in IEEE Communications Surveys & Tutorials, vol. 21, no. 2, pp. 1409-1434, Secondquarter 2019, doi: 10.1109/COMST.2018.2884835.

[27] A. Pelekanou, M. Anastasopoulos, A. Tzanakaki and D. Simeonidou, "Provisioning of 5G services employing machine learning techniques," *2018 International Conference on Optical Network Design and Modeling (ONDM)*, Dublin, 2018, pp. 200-205, doi: 10.23919/ONDM.2018.8396131.

[28] A. Shahraki, A. Taherkordi, Ø. Haugen and F. Eliassen, "A Survey and Future Directions on Clustering: From WSNs to IoT and Modern Networking Paradigms," in IEEE Transactions on Network and Service Management, doi: 10.1109/TNSM.2020.3035315.

[29] A. Solano and L. M. Contreras, "Information Exchange to Support Multi-Domain Slice Service Provision for 5G/NFV," *2020 IFIP Networking Conference (Networking)*, Paris, France, 2020, pp. 773-778.

[30] A. Tzanakaki *et al*., "5G infrastructures supporting end-user and operational services: The 5G-XHaul architectural perspective," *2016 IEEE International Conference on Communications Workshops (ICC)*, Kuala Lumpur, Malaysia, 2016, pp. 57-62, doi: 10.1109/ICCW.2016.7503764.

[31] A. Tzanakaki et al., "Wireless-Optical Network Convergence: Enabling the 5G Architecture to Support Operational and End-User Services", IEEE Commun. Mag., pp. 184 – 192, Aug. 2017

[32] A. Zappone, M. Di Renzo and M. Debbah, "Wireless Networks Design in the Era of Deep Learning: Model-Based, AI-Based, or Both?" in IEEE Transactions on Communications, vol. 67, no. 10, pp. 7331-7376, Oct. 2019, doi: 10.1109/TCOMM.2019.2924010.

[33] A. Zappone, M. Di Renzo, M. Debbah, T. T. Lam, X. Qian: Model-aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks for wireless system optimization, IEEE Vehicular Technology Magazine 14 (3), 60-69.

[34] Abeshu, A., & Chilamkurti, N. (2018). Deep learning: the frontier for distributed attack detection in fog-to-things computing. *IEEE Communications Magazine*, *56*(2), 169-175.

[35] Ahmad, I., Kumar, T., Liyanage, M., Okwuibe, J., Ylianttila, M., & Gurtov, A. (2018). Overview of 5G security challenges and solutions. *IEEE Communications Standards Magazine*, *2*(1), 36-43.

[36] Aldossari, Saud; Chen, Kwang-Cheng (2020). Predicting the Path Loss of Wireless Channel Models Using Machine Learning Techniques in MmWave Urban Communications.

[37] Aldossari, Saud; Chen, Kwang-Cheng. (2019). Machine Learning for Wireless Communication Channel Modeling: An Overview. Wireless Personal Communications. 106. 10.1007/s11277-019-06275-4.

[38] Alevizos, E., Artikis, A., & Paliouras, G. (2017). Event forecasting with pattern markov chains. In Proceedings of the 11th ACM International Conference on Distributed and Event-based Systems (pp. 146-157).

[39] Alevizos, E., Skarlatidis, A., Artikis, A., & Paliouras, G. (2017). Probabilistic complex event recognition: A survey. *ACM Computing Surveys (CSUR)*, *50*(5), 1-31

[40] Anna Goldie and Azalia Mirhoseini, "Placement Optimization with Deep Reinforcement Learning", In: Proceedings of the 2020 International Symposium on Physical Design, 2020.

[41] Antevski, K., Martín-Pérez, J., Garcia-Saavedra, A., Bernardos, C.J., Li, X., Baranda, J., Mangues-Bafalluy, J., Martnez, R. and Vettori, L., 2020, June. A Q-learning strategy for federation of 5G services. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)* (pp. 1-6). IEEE.

[42] APEX introduction. Available at: https://docs.onap.org/en/dublin/submodules/policy/apex-pdp.git/docs/APEX-Introduction.html

[43] Artificial Intelligence for Physical-Layer Wireless Communications, web-link: https://ieeeaccess.ieee.org/special-sections-closed/artificial-intelligence-for-physical-layer-wireless-communications/.

[44] Artikis, A., Sergot, M., & Paliouras, G. (2014). An event calculus for event recognition. *IEEE Transactions on Knowledge and Data Engineering*, *27*(4), 895-908

[45] Ateniese, G., Mancini, L. V., Spognardi, A., Villani, A., Vitali, D., & Felici, G. (2015). Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers. International Journal of Security and Networks, 10(3), 137-150.

[46] B. Makki, K. Chitti, A. Behravan, M.-S. Alouini, "A Survey of NOMA: Current Status and Open Research Challenges", IEEE Open Journal of the Communications Society (Volume: 1), pp. 179-189, Jan. 2020.

[47] B. Pan, F. Yan, X. Xue, E. Magelhaes and N. Calabretta, "Performance assessment of a fast optical add-drop multiplexer-based metro access network with edge computing," Journal of Optical Communications and Networking, vol. 11, no. 12, pp. 636-646, Dec. 2019.

[48] B. Uščumlić, I. Cerutti, A. Gravey, P. Gravey, D. Barth, M. Morvan and P. Castoldi, "Optimal dimensioning of the WDM unidirectional ECOFRAME optical packet ring," Photonic Network Communications, vol. 22, pp. 254-265, July 2011.

[49] Bai, Lu; Wang, Cheng-Xiang; Huang, Jie; Xu, Qian & Yang, Yuqian & Goussetis, George & Sun, Jian & Zhang, Wensheng. (2018). Predicting Wireless MmWave Massive MIMO Channel Characteristics Using Machine Learning Algorithms. Wireless Communications and Mobile Computing. 2018. 1-12. 10.1155/2018/9783863.

[50] Balevi, Eren, et al. "High Dimensional Channel Estimation Using Deep Generative Networks." *IEEE Journal on Selected Areas in Communications* 39.1 (2020): 18-30.

[51] Batalla, J. M., Andrukiewicz, E., Gomez, G. P., Sapiecha, P., Mavromoustakis, C. X., Mastorakis, G., & Imran, M. (2020). Security risk assessment for 5G networks: National perspective. *IEEE Wireless Communications*, *27*(4), 16-22.

[52] Bentaleb, Abdelhak, et al. "A survey on bitrate adaptation schemes for streaming media over HTTP." *IEEE Communications Surveys & Tutorials* 21.1 (2018): 562-585.

[53] Busoniu, L., Babuska, R., & De Schutter, B. (2008). A comprehensive survey of multiagent reinforcement learning. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38(2), 156-172.

[54] C. Benzaid and T. Taleb, “AI for Beyond 5G Networks: A Cyber-Security Defense or Offense Enabler?” in IEEE Network Magazine, Vol. 34, No. 6, Nov. 2020, pp. 140 - 147.

[55] C. Benzaid and T. Taleb, “ZSM Security: Threat Surface and Best Practices,” in IEEE Network Magazine, Vol. 34, No. 3, Jun. 2020, pp. 124 – 133

[56] C. Benzaid, M. Boukhalfa, and T. Taleb, “Robust Self-Protection Against Application-Layer (D)DoS Attacks in SDN Environment,” in Proc. IEEE WCNC 2020, Seoul, Korea, Apr. 2020

[57] C. Garcia Cifuentes, J. Issac, M. Wüthrich, S. Schaal, and J. Bohg, “Probabilistic Articulated Real-Time Tracking for Robot Manipulation,” IEEE Robot. Autom. Lett., vol. 2, no. 2, pp. 577–584, 2017, doi: 10.1109/LRA.2016.2645124

[58] C. Papagianni, J. Mangues, P. Bermudez, S. Barmpounakis, D. De Vleeschauwer, J. Brenes, E. Zeydan, C. Casetti, C. Guimarães, P. Murillo, A. Garcia-Saavedra, D. Corujo, T. Pepe, 5Growth: AI-driven 5G for Automation in Vertical Industries, in Proceedings of EuCNC 2020, Online Event, 15-18 June 2020.

[59] C. Papagianni, K. De Schepper, “PI2 for P4: An Active Queue Management Scheme for Programmable Data Planes”, ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT) 2019 Posters, December 2019

[60] C. Qi, Y. Hua, R. Li, Z. Zhao and H. Zhang, "Deep Reinforcement Learning With Discrete Normalized Advantage Functions for Resource Management in Network Slicing," in IEEE Communications Letters, vol. 23, no. 8, pp. 1337-1341, Aug. 2019, doi: 10.1109/LCOMM.2019.2922961

[61] C. Simonneau, J. C. Antona and D. Chiaroni, “Packet-optical Add/Drop multiplexer technology: A pragmatic way to introduce optical packet switching in the next generation of metro networks,” in Proc. 2009 IEEE LEOS Annual Meeting Conference, Belek-Antalya, Turkey, Oct. 2009.

[62] C. V. Forecast, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017--2022 White Paper," Feb, 2019.

[63] C. Zhang, P. Patras and H. Haddadi, "Deep Learning in Mobile and Wireless Networking: A Survey," in IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2224-2287, third quarter 2019, doi: 10.1109/COMST.2019.2904897

[64] Carlini, N., Liu, C., Kos, J., Erlingsson, Ú., & Song, D. (2018). The secret sharer: Measuring unintended neural network memorization & extracting secrets.

[65] Chaparadza, R. Requirements for a Generic Autonomic Network Architecture (GANA), suitable for Standardizable Autonomic Behaviour Specifications of Decision-Making-Elements (DMEs) for Diverse Networking Environments. Int. Eng. Consortium (IEC) Annu. Rev. Commun. 2008, 61, 165–194.

[66] Chen, M., Semiari, O., Saad, W., Liu, X., & Yin, C. (2019). Federated echo state learning for minimizing breaks in presence in wireless virtual reality networks. *IEEE Transactions on Wireless Communications*, *19*(1), 177-191.

[67] Chen, Mingzhe, et al. "Artificial neural networks-based machine learning for wireless networks: A tutorial." IEEE Communications Surveys & Tutorials 21.4 (2019): 3039-3071.

[68] Chergui, Hatim, and Christos Verikoukis. "Offline SLA-constrained deep learning for 5G networks reliable and dynamic end-to-end slicing." IEEE Journal on Selected Areas in Communications 38.2 (2019): 350-360.

[69] Cisco Annual Internet Report (2018–2023) White Paper. March 9, 2020, Online https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[70] Collobert, Ronan, and Samy Bengio. "Links between perceptrons, MLPs and SVMs." *Proceedings of the twenty-first international conference on Machine learning*. 2004.

[71] Cortes, Corinna, et al. "Adanet: Adaptive structural learning of artificial neural networks." *International conference on machine learning*. PMLR, 2017.

[72] Cristina Marquez, Marco Gramaglia, Marco Fiore, Albert Banchs, and Xavier Costa-Perez. 2018. How Should I Slice My Network? A Multi-Service Empirical Evaluation of Resource Sharing Efficiency. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MobiCom '18*). Association for Computing Machinery, New York, NY, USA, 191–206. DOI:https://doi.org/10.1145/3241539.3241567

[73] Cugola, G., & Margara, A. (2012). Processing flows of information: From data stream to complex event processing. *ACM Computing Surveys (CSUR)*, *44*(3), 1-62

[74] D. Andrei, H.-H. Yen, M. Tornatore, C. U. Martel and B. Mukherjee, “Integrated Provisioning of Sliding Scheduled Services Over WDM Optical Networks [Invited],” Journal of Optical Communications and Networking, vol. 1, no. 2, pp. A94-A105, July 2009.

[75] D. Arumugam1, J. Ki Lee, S. Saskin, M. Littman, Deep Reinforcement Learning from Policy-Dependent Human Feedback, arXiv preprint arXiv:1902.04257, 2019

[76] D. Bega, M. Gramaglia, M. Fiore, A. Banchs and X. Costa-Perez, "α-OMC: Cost-Aware Deep Learning for Mobile Network Resource Orchestration," IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Paris, France, 2019, pp. 423-428, doi: 10.1109/INFCOMW.2019.8845178

[77] D. Bega, M. Gramaglia, M. Fiore, A. Banchs and X. Costa-Pérez, "DeepCog: Optimizing Resource Provisioning in Network Slicing With AI-Based Capacity Forecasting," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 2, pp. 361-376, Feb. 2020, doi:10.1109/JSAC.2019.2959245.

[78] D. Bega, M. Gramaglia, R. Perz, M. Fiore, A. Banchs, and X. Costa-Perez. AI-Based Autonomous Control, Management, and Orchestration in 5G: from Standard to Algorithms in IEEE NEtwork

[79] D. Dey, A. v. Bochove, A. Koonen, D. Geuzebroek and M. Salvador, "FLAMINGO: a packet-switched IP-over-WDM all-optical MAN," in Proc. 27th European Conference on Optical Communication (ECOC), Amsterdam, Netherlands, Oct. 2001.

[80] D. Gündüz, P. de Kerret, N. D. Sidiropoulos, D. Gesbert, C. R. Murthy and M. van der Schaar, "Machine Learning in the Air," in IEEE Journal on Selected Areas in Communications, vol. 37, no. 10, pp. 2184-2199, Oct. 2019, doi: 10.1109/JSAC.2019.2933969

[81] D. P. Bertsekas, R. G. Gallager, P. Humblet, "Data networks", Vol. 2, Prentice-Hall International New Jersey, 1992

[82] Dai, Yueyue, et al. "Blockchain and deep reinforcement learning empowered intelligent 5G beyond." *IEEE Network* 33.3 (2019): 10-17.

[83] Draft Recommendation ITU-T Y.ML-IMT2020-E2E-MGMT, "Machine learning based end-to-end multi-domain network slice management and orchestration", 2020.

[84] Draft Recommendation ITU-T Y.ML-IMT2020-VNS, "Framework for network slicing management enabled by machine learning including input from verticals", 2020.

[85] Dreibholz, Thomas: "A 4G/5G Packet Core as VNF with Open Source MANO and OpenAirInterface", Proceedings of the 28th IEEE International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Dalmacija/Croatia, September 2020.

[86] Dreibholz, Thomas: "Flexible 4G/5G Testbed Setup for Mobile Edge Computing using OpenAirInterface and Open Source MANO", Proceedings of the 2nd International Workshop on Recent Advances for Multi-Clouds and Mobile Edge Computing (M2EC) in conjunction with the 34th International Conference on Advanced Information Networking and Applications (AINA), Campania/Italy, April 15, 2020.

[87] Dreibholz, Thomas; Mazumdar, Somnath; Zahid, Feroz; Taherkordi, Amirhosein; Gran, Ernst Gunnar: "Mobile Edge as Part of the Multi-Cloud Ecosystem: A Performance Study", Proceedings of the 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), Lombardia/Italy, February 13, 2019

[88] E. Bjornson and P. Giselsson, "Two Applications of Deep Learning in the Physical Layer of Communication Systems [Lecture Notes]," in IEEE Signal Processing Magazine, vol. 37, no. 5, pp. 134-140, Sept. 2020, doi: 10.1109/MSP.2020.2996545.

[89] E. N. Papasotiriou, A. A. Boulogeorgos and A. Alexiou, "Performance Analysis of THz Wireless Systems in the Presence of Antenna Misalignment and Phase Noise," in IEEE Communications Letters, vol. 24, no. 6, pp. 1211-1215, June 2020, doi: 10.1109/LCOMM.2020.2981336.

[90] E. N. Papasotiriou, J. Kokkoniemi, A. A. Boulogeorgos, J. Lehtomäki, A. Alexiou and M. Juntti, "A new look to 275 to 400 GHz band: Channel model and performance evaluation," 2018 IEEE 29th

Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Bologna, Italy, 2018, pp. 1-5, doi: 10.1109/PIMRC.2018.8580934.

[91] E. Pateromichelakis et al., "End-to-End Data Analytics Framework for 5G Architecture", IEEE Access, vol. 7, 2019, pp. 40295–40312

[92] E. Zeljković, N. Slamnik-Kriještorac, S. Latré and J. M. Marquez-Barja, "ABRAHAM: Machine Learning Backed Proactive Handover Algorithm Using SDN," in IEEE Transactions on Network and Service Management, vol. 16, no. 4, pp. 1522-1536, Dec. 2019, doi: 10.1109/TNSM.2019.2948883.

[93] Emami, M., Akbari, R., Javidan, R., & Zamani, A. (2019). A new approach for traffic matrix estimation in high load computer networks based on graph embedding and convolutional neural network. Transactions on Emerging Telecommunications Technologies, 30(6), e3604.

[94] Eren Balevi; Richard D. Gitlin, et al, Unsupervised Machine Learning in 5G Networks for Low Latency Communications, 2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC)

[95] ETSI Group Specification ENI 001, Experiential Networked Intelligence (ENI); ENI use cases, https://portal.etsi.org/webapp/workprogram/Report_WorkItem.asp?WKI_ID=54723#_blank

[96] ETSI Group Specification MEC 003 V2.1.1, Multi-access Edge Computing (MEC); Framework and Reference Architecture

[97] ETSI GS ENI 002 Experiential Networked Intelligence (ENI); ENI requirements. (n.d.). Retrieved from https://www.etsi.org/deliver/etsi_gs/ENI/001_099/002/02.01.01_60/gs_ENI002v020101p.pdf

[98] ETSI GS ENI 005, Experiential Networked Intelligence (ENI); System Architecture, https://www.etsi.org/deliver/etsi_gs/ENI/001_099/005/01.01.01_60/gs_ENI005v010101p.pdf

[99] ETSI TS 136 213 LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, V14.2.0 (2017-04).

[100] F. Mao, E. Blanco, M. Fu, R. Jain, A. Gupta, S. Mancel, R. Yuan, S. Guo, S. Kumar and Y. Tian, "Small Boxes Big Data: A Deep Learning Approach to Optimize Variable Sized Bin Packing," in 2017 IEEE Third International Conference on Big Data Computing Service and Applications (BigDataService), San Francisco, CA, USA, Apr. 2017

[101] F. Musumeci, C. Bellanzon, N. Carapellese, M. Tornatore, A. Pattavina, and S. Gosselin, "Optimal BBU Placement for 5G C-RAN Deployment Over WDM Aggregation Networks," J. Lightwave Technol. **34**, 1963-1970 (2016).

[102] F. Rezazadeh, H. Chergui and C. Verikoukis, "Continuous Multi-objective Zero-touch Network Slicing via Twin Delayed DDPG and OpenAI Gym," in Globecom 2020, Accepted.

[103] Ferreira, A. G. et al. (2017) 'Performance analysis of ToA-based positioning algorithms for static and dynamic targets with low ranging measurements', Sensors (Switzerland), 17(8), pp. 9–11. doi: 10.3390/s17081915.

[104] Fiandrino, C., Zhang, C., Patras, P., Banchs, A. and Widmer, J., 2020. A Machine Learning-based Framework for Optimizing the Operation of Future Networks. IEEE Communications Magazine.

[105] FlexRAN. https://networks.inf.ed.ac.uk/flexran/

[106] Fowler, R. J.; Paterson, M. S.; Tanimoto, S. L., "Optimal packing and covering in the plane are NP-complete", Information Processing Letters, 12 (3): 133–137. 1981.

[107] Fujimoto, S., van Hoof, H., and Meger, “Addressing function approximation error in actor-critic methods,” 2018, [Online] Available: https://arxiv.org/abs/1802.09477

[108] Gal, Y., Ghahramani, Z., Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, Proceedings of The 33rd International Conference on Machine Learning, in PMLR 48:1050-1059, 2016.

[109] Ganju, K., Wang, Q., Yang, W., Gunter, C. A., & Borisov, N. (2018). Property inference attacks on fully connected neural networks using permutation invariant representations. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (pp. 619-633).

[110] Gaur, M. (2020). Privacy Preserving Machine Learning Challenges and Solution Approach for Training Data in ERP Systems. International Journal of Computer Engineering and Technology.

[111] Giatrakos, N., Artikis, A., Deligiannakis, A., & Garofalakis, M. (2017). Complex event recognition in the big data era. *Proceedings of the VLDB Endowment*, *10*(12), 1996-1999

[112] Gomez, C.A., Wang, X. and Shami, A., 2019, December. Intelligent Active Queue Management Using Explicit Congestion Notification. In 2019 IEEE Global Communications Conference (GLOBECOM) (pp. 1-6). IEEE.

[113] Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems*. 2014.

[114] Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Methods for interpreting and understanding deep neural networks, Digital Signal Processing, Volume 73, 2018, Pages 1-15, ISSN 1051-2004, https://doi.org/10.1016/j.dsp.2017.10.011.

[115] Performance Isolation for 5G Network Slicing, https://5growth.eu/video-gallery/

[116] H. Qi and J. B. Moore, “Direct Kalman filtering approach for GPS/INS integration,” IEEE Trans. Aerosp. Electron. Syst., 2002, doi: 10.1109/TAES.2002.1008998

[117] H. Zang, J. P. Jue and B. Mukherjee, “A Review of Routing and Wavelength Assignment Approaches for Wavelength- Routed Optical WDM Networks,” Optical Networks Magazine, pp. 47-60, Jan. 2000.

[118] Hamidouche, K., Kasgari, A. T. Z., Saad, W., Bennis, M., & Debbah, M. (2018, May). Collaborative artificial intelligence (AI) for user-cell association in ultra-dense cellular systems. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)* (pp. 1-6). IEEE.

[119] https://developer.nvidia.com/cuda-zone

[120] https://docs.microsoft.com/en-us/cognitive-toolkit/

[121] https://eniwiki.etsi.org/index.php?title=ISG_ENI_Activities

[122] https://inform.tmforum.org/insights/2020/06/nwdaf-automating-the-5g-network-with-machine-learning-and-data-analytics/

[123] https://pypi.org/project/Theano/

[124] https://pytorch.org/

[125] https://spark.apache.org/mllib/

[126] https://www.tensorflow.org/

[127] Huang, Hongji, et al. "Deep learning for super-resolution channel estimation and DOA estimation based massive MIMO system." IEEE Transactions on Vehicular Technology 67.9 (2018): 8549-8560.

[128] Hughes, B., Bothe, S., Farooq, H., & Imran, A. (2019, February). Generative adversarial learning for machine learning empowered self organizing 5G networks. In *2019 International Conference on Computing, Networking and Communications (ICNC)* (pp. 282-286). IEEE.

[129] I. Goodfellow, J. Shlens y C. Szegedy, Explaining and harnessing adversarial examples, arXiv: 1412.6572, 2014.

[130] ITU-T Rec. Y.3172, Architectural framework for machine learning in future networks including IMT-2020, June 2019, retrieved from https://www.itu.int/rec/T-REC-Y.3172/en

[131] J. Bang, S. Oh, K. Kang and Y. Cho, "A Bayesian Regression Based LTE-R Handover Decision Algorithm for High-Speed Railway Systems," in IEEE Transactions on Vehicular Technology, vol. 68, no. 10, pp. 10160-10173, Oct. 2019, doi: 10.1109/TVT.2019.2935165.

[132] J. Baranda, J. Mangues, E. Zeydan, L. Vettori, R. Martínez, X. Li, A. Garcia-Saavedra, C. Fabiana Chiasserini, C. Casetti, K. Tomakh, O. Kolodiazhnyi, C. Jesus Bernardos, “On the Integration of AI/ML-based scaling operations in the 5Growth platform” , in Procs of the IEEE NFV-SDN 2020, Online Event, November 2020.

[133] J. Gil Herrera and J. F. Botero, "Resource Allocation in NFV: A Comprehensive Survey," in IEEE Transactions on Network and Service Management, vol. 13, no. 3, pp. 518-532, Sept. 2016, doi: 10.1109/TNSM.2016.2598420.

[134] J. Prados-Garzon, T. Taleb and M. Bagaa, "LEARNET: Reinforcement Learning Based Flow Scheduling for Asynchronous Deterministic Networks," *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Dublin, Ireland, 2020, pp. 1-6, doi: 10.1109/ICC40277.2020.9149092.

[135] J. Rauber, et. al., Foolbox: A python toolbox to benchmark the robustness of machine learning models, arXiv:1707.04131, 2018.

[136] J. S. Pujol Roig, D. M. Gutierrez-Estevez and D. Gündüz, "Management and Orchestration of Virtual Network Functions via Deep Reinforcement Learning," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 2, pp. 304-317, Feb. 2020, doi: 10.1109/JSAC.2019.2959263

[137] J. Wang et al., "A Machine Learning Framework for Resource Allocation Assisted by Cloud Computing," in IEEE Network, vol. 32, no. 2, pp. 144-151, March-April 2018, doi: 10.1109/MNET.2018.1700293.

[138] James, G.M. Variance and Bias for General Loss Functions. Machine Learning 51, 115–135 (2003). https://doi.org/10.1023/A:1022899518027

[139] Jan G. De Gooijer, Kuldeep Kumar, “Some recent developments in non-linear time series modelling, testing, and forecasting”, *International Journal of Forecasting*, Volume 8, Issue 2, 1992, Pages 135-156, ISSN 0169-2070, https://doi.org/10.1016/0169-2070(92)90115-P

[140] Jan G. De Gooijer, Rob J. Hyndman, “25 years of time series forecasting”, *International Journal of Forecasting*, Volume 22, Issue 3, 2006, Pages 443-473, ISSN 0169-2070, https://doi.org/10.1016/j.ijforecast.2006.01.001

[141] Jiang, D., Wang, X., Guo, L., Ni, H., & Chen, Z. (2011). Accurate estimation of large-scale IP traffic matrix. AEU-International Journal of Electronics and Communications, 65(1), 75-86.

[142] Jiang, J., Dun, C., Huang, T., & Lu, Z. (2018). Graph convolutional reinforcement learning. *arXiv preprint arXiv:1810.09202*.

[143] Jiang, Wei; Schotten, Hans. (2019). Neural Network-based Fading Channel Prediction: A Comprehensive Overview. IEEE Access. PP. 1-1. 10.1109/ACCESS.2019.2937588.

[144] Jose A. Ayala-Romero, Andres Garcia-Saavedra, Marco Gramaglia, Xavier Costa-Perez, Albert Banchs, and Juan J. Alcaraz. 2019. VrAIn: A Deep Learning Approach Tailoring Computing and Radio Resources in Virtualized RANs. In <i>The 25th Annual International Conference on Mobile Computing and Networking</i> (<i>MobiCom '19</i>). Association for Computing Machinery, New York, NY, USA, Article 30, 1–16. DOI:https://doi.org/10.1145/3300061.3345431

[145] Jose Mendes, Xian Jun Jiao, Andres Garcia-Saavedra, Felipe Huici, Ingrid Moerman, “Cellular access multi-tenancy through small-cell virtualization and common RF front-end sharing”, *Computer Communications*, Volume 133, 2019, Pages 59-66, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2018.10.010

[146] Juhoon Kim and Min Xie, *A study of slice-aware service assurance for network function virtualization*, IEEE Conference on network softwarization (NetSoft), 2019

[147] K. Shrikhande, I. White, D. Wonglumsom, S. Gemelos, M. Rogge, Y. Fukashiro, M. Avenarius and L. Kazovsky, “HORNET: a packet-over-WDM multiple access metropolitan area ring network,” IEEE Journal on Selected Areas in Communications, vol. 18, no. 10, pp. 2004-2016, Oct. 2000.

[148] Kakkavas, G., Tsitseklis, K., Karyotis, V., & Papavassiliou, S. (2020). A Software Defined Radio Cross-Layer Resource Allocation Approach for Cognitive Radio Networks: From Theory to Practice. IEEE Transactions on Cognitive Communications and Networking, 6(2), 740-755.

[149] Kalman, R. E. (1960) ‘A new approach to linear filtering and prediction problems’, Journal of Fluids Engineering, Transactions of the ASME. doi: 10.1115/1.3662552.

[150] Karjee, Jyotirmoy, Rajan Garg, and Vaishal Tijoriwala. "A Reinforcement Learning Approach to Handle Radio Link Failure in Elevator Scenario." 2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC). IEEE, 2020.

[151] Katzouris, N., & Artikis, A. WOLED: A Tool for Online Learning Weighted Answer Set Rules for Temporal Reasoning Under Uncertainty, in *Int. Conf. on Principles on Knowledge Representation & Reasoning, KR, 2020*

[152] Katzouris, N., Michelioudakis, E., Artikis, A., & Paliouras, G. (2018, September). Online learning of weighted relational rules for complex event recognition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 396-413). Springer, Cham

[153] Khan, R., Kumar, P., Jayakody, D. N. K., & Liyanage, M. (2019). A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements, and Future Directions. *IEEE Communications Surveys & Tutorials*, *22*(1), 196-248.

[154] Kindervag, J. (2010). No more chewy centers: Introducing the zero trust model of information security. Forrester Research.

[155] Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes." *arXiv preprint arXiv:1312.6114* (2013).

[156] Kubernetes. https://kubernetes.io/

[157] L. Chot Hun, O. Lee Yeng, L. Tien Sze, and K. Voon Chet, “Kalman Filtering and Its Real-Time Applications,” 2016, doi: 10.5772/62352

[158] L. Dittmann et al., “The European IST project DAVID: a viable approach toward optical packet switching,” IEEE Journal on Selected Areas in Communications, vol. 21, no. 7, pp. 1026-1040, Sept. 2003.

[159] L. Lei, D. Yuan, C. K. Ho and S. Sun, “Joint Optimization of Power and Channel Allocation with Non-Orthogonal Multiple Access for 5G Cellular Systems,” 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, 2015, pp. 1-6, doi: 10.1109/GLOCOM.2015.7417761

[160] L. Tang, X. He, P. Zhao, G. Zhao, Y. Zhou and Q. Chen, "Virtual Network Function Migration Based on Dynamic Resource Requirements Prediction," in IEEE Access, vol. 7, pp. 112348-112362, 2019, doi: 10.1109/ACCESS.2019.2935014.

[161] L. U. Khan, U. Majeed and C. S. Hong, "Federated Learning for Cellular Networks: Joint User Association and Resource Allocation," 2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS), Daegu, Korea (South), 2020, pp. 405-408, doi: 10.23919/APNOMS50412.2020.9237045.

[162] L. Zhang and N. Ansari, "Optimizing the Deployment and Throughput of DBSs for Uplink Communications," in IEEE Open Journal of Vehicular Technology, vol. 1, pp. 18-28, 2020, doi: 10.1109/OJVT.2019.2954390.

[163] Lea Skorin-Kapov, Martin Varela, Tobias Hossfeld, and Kuan-Ta Chen, *A survey of emerging concepts and challenges for QoE management of multimedia services*, ACM Transactions on multimedia computing, communications, and applications, Vol. 1, No. 1, Jan, 2017

[164] LeCun, Yann. "Generalization and network design strategies." Connectionism in perspective 19 (1989): 143-155.

[165] Li, Feng, et al. "Multi-armed-bandit-based spectrum scheduling algorithms in wireless networks: A survey." *IEEE Wireless Communications* 27.1 (2020): 24-30.

[166] Li, S. (2020). Zero trust based internet of things. *EAI Endorsed Transactions on Internet of Things*, *5*(20).

[167] Li, S., Xu, C., & Xie, M. (2012). A robust O (n) solution to the perspective-n-point problem. *IEEE transactions on pattern analysis and machine intelligence*, *34*(7), 1444-1450.

[168] Li, Y., Zhao, Y., Li, J., Zhang, J., Yu, X., & Zhang, J. (2019). Side channel attack-aware resource allocation for URLLC and eMBB slices in 5G RAN. *IEEE Access*, *8*, 2090-2099.

[169] Liang, Ke, et al. "Integrated prefetching and caching for adaptive video streaming over HTTP: an online approach." *Proceedings of the 6th ACM Multimedia Systems Conference*. 2015.

[170] Lillicrap, Timothy P and Hunt, Jonathan J and Pritzel, Alexander and Heess, Nicolas and Erez, Tom and Tassa, Yuval and Silver, David and Wierstra, Daan. Continuous control with deep reinforcement learning, arXiv preprint arXiv:1509.02971, 2015

[171] Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y. C., Yang, Q., ... & Miao, C. (2020). Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*.

[172] Lin, Y., Dai, X., Li, L., & Wang, F. Y. (2018). Pattern sensitive prediction of traffic flow based on generative adversarial framework. *IEEE Transactions on Intelligent Transportation Systems*, *20*(6), 2395-2400.

[173] Lin, Zinan, et al. "Generating High-fidelity, Synthetic Time Series Datasets with DoppelGANger." *arXiv preprint arXiv:1909.13403* (2019).

[174] Liu, Y., Peng, J., Kang, J., Iliyasu, A. M., Niyato, D., & El-Latif, A. A. A. (2020). A Secure Federated Learning Framework for 5G Networks. *arXiv preprint arXiv:2005.05752*.

[175] Lukoševičius M. (2012) A Practical Guide to Applying Echo State Networks. In: Montavon G., Orr G.B., Müller KR. (eds) Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science, vol 7700. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35289-8_36

[176] Luo, Changqing, et al. "Channel state information prediction for 5G wireless communications: A deep learning approach." *IEEE Transactions on Network Science and Engineering* (2018).

[177] M. Anastasopoulos et al., “ICT platforms in support of future railway systems”, in proc. of TRA 2018, Apr. 2018

[178] M. Anastasopoulos, A. Tzanakaki and D. Simeonidou, "Stochastic Energy Efficient Cloud Service Provisioning Deploying Renewable Energy Sources," in *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3927-3940, Dec. 2016.

[179] M. Chiang, S.H. Low, R.A. Calderbank, and J.C. Doyle, “Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures”, Proceedings of the IEEE, 95 (1), pp. 255-312, 2007.

[180] M. Di Renzo, A. Zappone, T. T. Lam, and M. Debbah, “Systemlevel modeling and optimization of the energy efficiency in cellular networks—A stochastic geometry framework,” IEEE Trans. Wireless Commun., vol. 17, no. 4, pp. 2539–2556, Apr. 2018.

[181] M. Diamanti, G. Fragkos, E. E. Tsiropoulou and S. Papavassiliou, "Unified User Association and Contract-Theoretic Resource Orchestration in NOMA Heterogeneous Wireless Networks," in IEEE Open Journal of the Communications Society, vol. 1, pp. 1485-1502, 2020, doi: 10.1109/OJCOMS.2020.3024778.

[182] M. E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," Journal of Machine Learning Research, 2001

[183] M. Fiorani, S. Tombaz, J. Martensson, B. Skubic, L. Wosinska and P. Monti, "Modeling energy performance of C-RAN with optical transport in 5G network scenarios," in *IEEE/OSA Journal of Optical Communications and Networking*, vol. 8, no. 11, pp. B21-B34, Nov. 2016.

[184] M. N. Patwary, S. Junaid Nawaz, M. A. Rahman, S. K. Sharma, M. M. Rashid and S. J. Barnes, "The Potential Short- and Long-Term Disruptions and Transformative Impacts of 5G and Beyond Wireless Networks: Lessons Learnt From the Development of a 5G Testbed Environment," in IEEE Access, vol. 8, pp. 11352-11379, 2020, doi: 10.1109/ACCESS.2020.2964673.

[185] Maimó, L. F., Gómez, Á. L. P., Clemente, F. J. G., Pérez, M. G., & Pérez, G. M. (2018). A self-adaptive deep learning-based system for anomaly detection in 5G networks. *IEEE Access*, *6*, 7700-7712.

[186] Maksymyuk, T., Gazda, J., Luntovskyy, A., & Klymash, M. (2018, September). Artificial Intelligence based 5G Coverage Design and optimization using Deep Generative Adversarial Neural

Networks. In *2018 International Conference on Information and Telecommunication Technologies and Radio Electronics (UkrMiCo)* (pp. 1-4). IEEE.

[187] Manuel Lopez-Martin, Belen Carro, Jaime Lloret, Santiago Egea and Antonio Sanchez-Esguevillas, *Deep Learning model for multimedia quality of experience prediction based on network flow packets*, IEEE Communications Magazine, Sep 2018

[188] Mao, Bomin, et al. "A novel non-supervised deep-learning-based network traffic control method for software defined wireless networks." IEEE Wireless Communications 25.4 (2018): 74-81

[189] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In Artificial Intelligence and Statistics (pp. 1273-1282). PMLR.

[190] Megiddo, Nimrod; Tamir, Arie, "On the complexity of locating linear facilities in the plane", Operations Research Letters, 1(5): 194–197. 1982

[191] Mehrabi, Mehrtash, et al. "Decision Directed Channel Estimation Based on Deep Neural Network $ k $-Step Predictor for MIMO Communications in 5G." IEEE Journal on Selected Areas in Communications 37.11 (2019): 2443-2456.

[192] MELODIC project, online: https://melodic.cloud/

[193] Michailidis, E. T., Nomikos, N., Trakadas, P., & Kanatas, A. G. (2019). Three-dimensional modeling of mmWave doubly massive MIMO aerial fading channels. *IEEE Transactions on Vehicular Technology*, *69*(2), 1190-1202

[194] Min Xie, Joan Pujol-Roig, Foivos Michelinakis, Thomas Dreibholz, Carmen Guerrero, Adrisn Gallego Sanchez, Wint Yi Poe, Yue Wang, and Alhmed Elmokashfi, "*AI-driven closed-loop service assurance with service exposure*", European conference on networks and communications (EuCNC), 2020

[195] Min Xie, Qiong Zhang, Andres Gonzalez, Pål Grønsund, Paparao Palacharla, Tadashi Ikeuchi, "*Service assurance in 5G networks: a study of joint monitoring and analytics*", 30th IEEE annual international symposium on personal, indoor, and mobile radio communications (PIMRC), 2019

[196] Mirshghallah, F., Taram, M., Vepakomma, P., Singh, A., Raskar, R., & Esmaeilzadeh, H. (2020). Privacy in Deep Learning: A Survey. arXiv preprint arXiv:2004.12254.

[197] Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015). https://doi.org/10.1038/nature14236

[198] Mpeg, I. *Information technology-dynamic adaptive streaming over http (dash)-part 1: media presentation description and segment formats*. ISO. IEC MPEG, Tech. Rep, 2012.

[199] Muhammad Fahad Khan, Kok-Lim Alvin Yau, Rafidah MD. Noor, Muhammad Ali Imran, Survey and taxonomy of clustering algorithms in 5G, Journal of Network and Computer Applications, Volume 154, 2020, 102539, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2020.102539.

[200] N. Calabretta, W. Miao, K. Mekonnen and K. Prifti, "SOA Based Photonic Integrated WDM Cross-Connects for Optical Metro-Access Networks," Applied Sciences, vol. 7, no. 9, p. 865, Aug. 2017.

[201] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 2017, pp. 39-57, doi: 10.1109/SP.2017.49.

[202] N. Gkatzios, M. Anastasopoulos, A. Tzanakaki and D. Simeonidou, "Compute Resource Disaggregation: An Enabler for Efficient 5G RAN Softwarisation," 2018 European Conference on Networks and Communications (EuCNC), Ljubljana, Slovenia, 2018, pp. 1-5, doi: 10.1109/EuCNC.2018.8443270.

[203] N. Papernot, et. al., Technical Report on the CleverHans v2.1.0 Adversarial Examples Library, arXiv:1610.00768., 2018.

[204] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik and A. Swami, "The Limitations of Deep Learning in Adversarial Settings," 2016 IEEE European Symposium on Security and Privacy (EuroS&P), Saarbruecken, Germany, 2016, pp. 372-387, doi: 10.1109/EuroSP.2016.36.

[205] N. Zhao, Y. Liang, D. Niyato, Y. Pei, M. Wu and Y. Jiang, "Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks," in IEEE Transactions on Wireless Communications, vol. 18, no. 11, pp. 5141-5152, Nov. 2019, doi: 10.1109/TWC.2019.2933417

[206] NGMN, 5G White Paper 2, accessible at: https://www.ngmn.org/work-programme/5g-white-paper-2.html

[207] Nguyen, T. T., & Reddi, V. J. (2019). Deep reinforcement learning for cyber security. *arXiv preprint arXiv:1906.05799*.

[208] Nie, L., Jiang, D., Guo, L., & Yu, S. (2016). Traffic matrix prediction and estimation based on deep learning in large-scale IP backbone networks. Journal of Network and Computer Applications, 76, 16-22.

[209] Nie, L., Jiang, D., Guo, L., Yu, S., & Song, H. (2016, December). Traffic matrix prediction and estimation based on deep learning for data center networks. In 2016 IEEE Globecom Workshops (GC Wkshps) (pp. 1-6). IEEE.

[210] Nokia (white paper), "How 5G is bringing an energy efficiency revolution", online [available] https://onestore.nokia.com/asset/207360?_ga=2.71882131.1671284638.1604662646-2037315745.1604662646.

[211] Norberto Garcia, Tomas Alcaniz, Aurora González-Vidal, Jorge Bernal Bernabe, Diego Rivera, Antonio Skarmeta, Distributed real-time SlowDoS attacks detection over encrypted traffic using Artificial Intelligence, Journal of Network and Computer Applications, Volume 173, 2021, 102871, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2020.102871.

[212] Oddi, Guido, et al. "A proactive link-failure resilient routing protocol for MANETs based on reinforcement learning." 2012 20th Mediterranean Conference on Control & Automation (MED). IEEE, 2012

[213] Open Source MANO (OSM), online: https://osm.etsi.org/

[214] OpenAirInterface. https://www.openairinterface.org/

[215] OpenStack. https://www.openstack.org/

[216] O-RAN Alliance, “O-RAN Working Group 2. AI/ML workflow description and requirements (O-RAN.WG2.AIML-v01.01),” Technical Report, 2020.

[217] O-RAN Alliance, “O-RAN-WG1-O-RAN Architecture Description -v02.00.00.” Technical Specification, Jul. 2020

[218] O-RAN Alliance, Working Group 1, O-RAN Operations and Maintenance Architecture, Technical Specification O-RAN.WG1.OAM-Architecture-v03.00, April 2020.

[219] O-RAN Alliance, Working Group 1, Use Cases Detailed Specification, Technical Specification O-RAN.WG1.Use-Cases-Detailed-Specification-v03.00, July 2020

[220] O-RAN Bronze Release. https://www.o-ran.org/software

[221] P. Calégari, F. Guidec, P. Kuonen, D. Kobler, “Parallel Island-Based Genetic Algorithm for Radio Network Design”, Journal of Parallel and Distributed Computing (JPDC) – Special Issue on Parallel Evolutionary Computing, Vol. 47 N.1, Pages 86-90 1997.

[222] P. Chou, W. Chen, C. Wang, R. Hwang and W. Chen, "Deep Reinforcement Learning for MEC Streaming with Joint User Association and Resource Management," ICC 2020 - 2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 2020, pp. 1-7, doi: 10.1109/ICC40277.2020.9149086.

[223] P. J. Werbos, “Backpropagation through time: What it does and how to do it,” Proc. IEEE, vol. 78, no. 10, pp. 1550–1560, Oct. 1990.

[224] P. S. Maybeck and G. M. Siouris, “Stochastic Models, Estimation, and Control, Volume I,” IEEE Transactions on Systems, Man and Cybernetics. 1980, doi: 10.1109/TSMC.1980.4308494

[225] Panagis Magdalinos, Sokratis Barmpounakis, Panagiotis Spapis, Alexandros Kaloxylos, Georgios Kyprianidis, Apostolis Kousaridas, Nancy Alonistioti, Chan Zhou, A context extraction and profiling engine for 5G network resource mapping, Computer Communications, Volume 109, 2017, Pages 184-201, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2017.06.003.

[226] Papernot, Nicolas; McDaniel, Patrick; Sinha, Arunesh; and Wellman, Michael. SoK: Towards the Science of security and privacy in machine learning. (2018). Proceedings of 3rd IEEE European Symposium on Security and Privacy (EuroS&P), London, 2018 April 24-26. 1-19. Research Collection School Of Computing and Information Systems.

[227] Partala, Juha & Lovén, Lauri & Peltonen, Ella & Porambage, Pawani & Ylianttila, Mika & Seppänen, Tapio. (2019). EdgeAI: A Vision for Privacy-preserving Machine Learning on the Edge.

[228] Paulo Valente Klaine and others, “A survey of machine learning techniques applied to self-organizing cellular networks”, IEEE Communications Surveys & Tutorials, Vol. 19, No.4, 2017

[229] Peken, Ture, Ravi Tandon, and Tamal Bose. "Unsupervised mmWave Beamforming via Autoencoders." *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020.

[230] Project 5G-CLARITY - Beyond 5G multi-tenant private networks integrating cellular, Wi-Fi, and LiFi, powered by artificial intelligence and intent-based policy, Online: https://www.5gclarity.com/

[231] Project 5G-COMPLETE - A unified network, computational and storage resource management framework targeting end-to-end performance optimization for secure 5G multi-technology and multi-tenancy environments, Online: https://5gcomplete.eu/

[232] Project 5G-COMPLETE deliverable D2.1 “Initial report on 5G-COMPLETE network architecture, interfaces and supported functions”, https://5gcomplete.eu/public-deliverables/

[233] Project 5G-COMPLETE deliverable D4.2 “Initial report on the HW implementation of AI and ML techniques in support of 5G services”, https://5gcomplete.eu/public-deliverables/

[234] Project 5GENESIS - 5th Generation End-to-end Network, Experimentation, System Integration, and Showcasing, Online: https://5genesis.eu/

[235] Project 5GENESIS, Deliverable D3.1, “Management and Orchestration (Release A)”. Available at: https://5genesis.eu/wp-content/uploads/2019/10/5GENESIS_D3.1_v1.0.pdf

[236] Project 5GENESIS, Deliverable D3.5, “Monitoring and Analytics (Release A)”, online: https://5genesis.eu/wp-content/uploads/2019/10/5GENESIS_D3.5_v1.0.pdf

[237] Project 5GENESIS, Deliverable D6.1 “Trials and Experimentations – Cycle 1” [Online], https://5genesis.eu/wp-content/uploads/2019/08/5GENESIS_D6.1_v1.00.pdf, Accessed: Oct. 2020.

[238] Project 5GROWH - 5G-enabled Growth in Vertical Industries, Online: https://5growth.eu/

[239] Project 5GROWTH Deliverable D2.2 “Initial implementation of 5G End-to-End Service Platform”, online: https://5growth.eu/wp-content/uploads/2020/05/D2.2-Initial_implementation_of_5G_End-to-End_Service_Platform.pdf

[240] Project 5G-SOLUTIONS for European citizens, online: https://www.5gsolutionsproject.eu/

[241] Project 5G-TRANSFORMER Deliverable D5.4 “Report on trials results”, Online: http://5g-transformer.eu/wp-content/uploads/2019/11/D5.4-5G-TRANSFORMER_Report_on_trials_results.pdf

[242] Project ARIADNE - Artificial Intelligence Aided D-Band Network for 5G Long Term Evolution, online: https://www.ict-ariadne.eu/

[243] Project INSPIRE-5Gplus Deliverable D5.1 “5G Security Test Cases”, Online: https://doi.org/10.5281/zenodo.4569524

[244] Project INSPIRE-5Gplus, Intelligent Security and Pervasive Trust for 5G and Beyond, Online: https://www.inspire-5gplus.eu/

[245] Project SliceNet - End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks, Online: https://slicenet.eu/

[246] Q. Chen, S. Zhang, S. Xu and S. Cao, "Efficient MIMO Detection with Imperfect Channel Knowledge - A Deep Learning Approach," 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 2019, pp. 1-6, doi: 10.1109/WCNC.2019.8885582.

[247] Q. Wang et al., "SliceNet: End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks," 2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Valencia, Spain, 2018, pp. 1-5, doi: 10.1109/BMSB.2018.8436800.

[248] Quang Tran Anh Pham, Yassine Hadjadj-Aoul, Abdelkader Outtagarts. Deep Reinforcement Learning based QoS-aware Routing in Knowledge-defined networking. Qshine 2018 - 14th EAI International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Dec 2018, Ho Chi Minh City, Vietnam. pp.1-13. ffhal-01933970f

[249] R. E. Kalman, “A new approach to linear filtering and prediction problems,” J. Fluids Eng. Trans. ASME, vol. 82, no. 1, pp. 35–45, 1960, doi: 10.1115/1.3662552

[250] R. Li, Z. Zhao, X. Chen, J. Palicot and H. Zhang, "TACT: A Transfer Actor-Critic Learning Framework for Energy Saving in Cellular Radio Access Networks," in IEEE Transactions on Wireless Communications, vol. 13, no. 4, pp. 2000-2011, April 2014, doi: 10.1109/TWC.2014.022014.130840

[251] R. Loftin et al., "Learning something from nothing: Leveraging implicit human feedback strategies," The 23rd IEEE International Symposium on Robot and Human Interactive Communication, Edinburgh, UK, 2014, pp. 607-612, doi: 10.1109/ROMAN.2014.6926319.

[252] Raedt, L. D., Kersting, K., Natarajan, S., & Poole, D. (2016). Statistical relational artificial intelligence: Logic, probability, and computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *10*(2), 1-189

[253] Restuccia, F., & Melodia, T. (2020). Physical-Layer Deep Learning: Challenges and Applications to 5G and Beyond. *arXiv preprint arXiv:2004.10113*.

[254] Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero Trust Architecture (No. NIST Special Publication (SP) 800-207). National Institute of Standards and Technology.

[255] S. Barbounakis, A. Kaloxylos, et. al., Data Analytics for 5G Networks: A complete framework for network access selection and traffic steering, International Journal On Advances in Telecommunications, v 11 n 3&4 2018

[256] S. Dutta, S. Gera, A. Verma and B. Viswanathan, "SmartScale: Automatic Application Scaling in Enterprise Clouds," 2012 IEEE Fifth International Conference on Cloud Computing, Honolulu, HI, 2012, pp. 221-228, doi: 10.1109/CLOUD.2012.12

[257] S. Maranò, W. M. Gifford, H. Wymeersch and M. Z. Win, "NLOS Identification and Mitigation for Localization," IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, 2010

[258] S. Mazuelas, A. Conti, J. C. Allen and M. Z. Win, "Soft Range Information for Network Localization," in IEEE Transactions on Signal Processing, vol. 66, no. 12, pp. 3155-3168, 15 June15, 2018, doi: 10.1109/TSP.2018.2795537

[259] S. Rahman, T. Ahmed, M. Huynh, M. Tornatore and B. Mukherjee, "Auto-Scaling VNFs Using Machine Learning to Improve QoS and Reduce Cost," 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, 2018, pp. 1-6, doi: 10.1109/ICC.2018.8422788

[260] S. Sevgican, M. Turan, K. Gökarslan, H. B. Yilmaz and T. Tugcu, "Intelligent network data analytics function in 5G cellular networks using machine learning," in Journal of Communications and Networks, vol. 22, no. 3, pp. 269-280, June 2020, doi: 10.1109/JCN.2020.000019.

[261] S. Wang and T. Lv, "Deep Reinforcement Learning for Demand-Aware Joint VNF Placement-and-Routing," 2019 IEEE Globecom Workshops (GC Wkshps), Waikoloa, HI, USA, 2019, pp. 1-6, doi: 10.1109/GCWkshps45667.2019.9024688.

[262] S. Yang, F. Li, S. Trajanovski, R. Yahyapour and X. Fu, "Recent Advances of Resource Allocation in Network Function Virtualization," in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 2, pp. 295-314, 1 Feb. 2021, doi: 10.1109/TPDS.2020.3017001.

[263] Sahil Garg, Kuljeet Kaur, Shalini Batra, Georges Kaddoum, Neeraj Kumar, Azzedine Boukerche, A multi-stage anomaly detection scheme for augmenting the security in IoT-enabled applications, Future Generation Computer Systems, Volume 104, 2020, Pages 105-118, ISSN 0167-739X, https://doi.org/10.1016/j.future.2019.09.038.

[264] Saho, K. (2018) ‘Kalman Filter for Moving Object Tracking: Performance Analysis and Filter Design’, Kalman Filters - Theory for Advanced Applications. doi: 10.5772/intechopen.71731

[265] Sakurada, Mayu, and Takehisa Yairi. "Anomaly detection using autoencoders with nonlinear dimensionality reduction." *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*. 2014.

[266] Samek, W., Stanczak, S., and Wiegand, T., “The Convergence of Machine Learning and Communications”, 2017.

[267] Sampaio, Lauren SR, et al. "Using NFV and reinforcement learning for anomalies detection and mitigation in SDN." 2018 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2018

[268] Sarwate, A. D., & Chaudhuri, K. (2013). Signal processing and machine learning with differential privacy: Algorithms and challenges for continuous data. IEEE signal processing magazine, 30(5), 86-94.

[269] Silver, D., Schrittwieser, J., Simonyan, K. *et al.* Mastering the game of Go without human knowledge. *Nature* **550,** 354–359 (2017). https://doi.org/10.1038/nature24270

[270] Soltani, Mehran; Mirzaei, Ali; Pourahmadi, Vahid and Sheikhzadeh, Hamid. (2018). Deep Learning-Based Channel Estimation.

[271] Stamou A, Kakkavas G, Tsitseklis K, Karyotis V, Papavassiliou S. Autonomic Network Management and Cross-Layer Optimization in Software Defined Radio Environments. Future Internet. 2019; 11(2):37. https://doi.org/10.3390/fi11020037

[272] Stamou, A., Kakkavas, G., Tsitseklis, K., Karyotis, V., & Papavassiliou, S. (2019). Autonomic network management and cross-layer optimization in software defined radio environments. Future Internet, 11(2), 37.

[273] Sutton, Richard S and Barto, Andrew G, “Reinforcement learning: An introduction”, 2018, MIT press, ISBN 978-0262039246

[274] Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[275] T. Lillicrap et al., ``Continuous control with deep reinforcement learning,´´ in ICRL, 2016, [Online] Available: https://arxiv.org/abs/1509.02971.

[276] T. Pfeiffer, "Next generation mobile fronthaul and midhaul architectures [Invited]," in *IEEE/OSA Journal of Optical Communications and Networking*, vol. 7, no. 11, pp. B38-B45, November 1 2015.

[277] T. Van Nguyen, Y. Jeong, H. Shin and M. Z. Win, "Machine Learning for Wideband Localization," IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, vol. 14, 2015.

[278] T. Wang, C. Wen, H. Wang, F. Gao, T. Jiang and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," in China Communications, vol. 14, no. 11, pp. 92-111, Nov. 2017, doi: 10.1109/CC.2017.8233654.

[279] T. X. Tran, A. Younis and D. Pompili, "Understanding the Computational Requirements of Virtualized Baseband Units Using a Programmable Cloud Radio Access Network Testbed," *2017 IEEE International Conference on Autonomic Computing (ICAC)*, Columbus, OH, 2017, pp. 221-226, doi: 10.1109/ICAC.2017.42

[280] Thing, Vrizlynn LL. "IEEE 802.11 network anomaly detection and attack classification: A deep learning approach." *2017 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2017.

[281] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, Daan Wierstra, "Continuous control with deep reinforcement learning", 2015, arXiv preprint, https://arxiv.org/abs/1509.02971

[282] uni|MS – Unified Network Management Product Brochure: Network Lifecycle Management Redefined!, Intracom Telecom, [Online]: http://www.intracom-telecom.com/downloads/pdf/products/netw_man/uniMS.pdf

[283] V. Yajnanarayana, H. Rydén and L. Hévizi, "5G Handover using Reinforcement Learning," 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 2020, pp. 349-354, doi: 10.1109/5GWF49715.2020.9221072.

[284] Vinchoff, C., Chung, N., Gordon, T., Lyford, L., & Aibin, M. (2020, July). Traffic Prediction in Optical Networks Using Graph Convolutional Generative Adversarial Networks. In *2020 22nd International Conference on Transparent Optical Networks (ICTON)* (pp. 1-4). IEEE.

[285] W. Kellerer, P. Kalmbach, A. Blenk, A. Basta, M. Reisslein and S. Schmid, "Adaptable and Data-Driven Softwarized Networks: Review, Opportunities, and Challenges," in Proceedings of the IEEE, vol. 107, no. 4, pp. 711-731, April 2019, doi: 10.1109/JPROC.2019.2895553.

[286] W. Miao, J. v. Weerdenburg, R. v. Uden, H. d. Waardt, T. Koonen, C. Okonkwo and N. Calabretta, "Low Latency Optical Label Switched Add-Drop Node for Multi-Tb/s Data Center Interconnect Metro Networks," in Proc. 42nd European Conference on Optical Communication (ECOC), Dusseldorf, Germany, Sept. 2016.

[287] Wang, Zhilong, et al. "Failure prediction using machine learning and time series in optical network." Optics Express 25.16 (2017): 18553-18565.

[288] Welch, G. and Bishop, G. (2006) 'An Introduction to the Kalman Filter', In Practice, 7(1), pp. 1–16. doi: 10.1.1.117.6808.

[289] Welch, G. F. (2009) 'HISTORY: The use of the Kalman filter for human motion tracking in virtual reality', Presence: Teleoperators and Virtual Environments, 18(1), pp. 72–91. doi: 10.1162/pres.18.1.72.

[290] Wold, Svante, Kim Esbensen, and Paul Geladi. "Principal component analysis." Chemometrics and intelligent laboratory systems 2.1-3 (1987): 37-52.

[291] Wu, X., Fredrikson, M., Jha, S., & Naughton, J. F. (2016). A methodology for formalizing model-inversion attacks. In 2016 IEEE 29th Computer Security Foundations Symposium (CSF) (pp. 355-370). IEEE.

[292] X. Chen et al., "Multi-Tenant Cross-Slice Resource Orchestration: A Deep Reinforcement Learning Approach," in IEEE Journal on Selected Areas in Communications, vol. 37, no. 10, pp. 2377-2392, Oct. 2019, doi: 10.1109/JSAC.2019.2933893

[293] X. Chen, H. Zhang, C. Wu, S. Mao, Y. Ji and M. Bennis, "Optimized Computation Offloading Performance in Virtual Edge Computing Systems Via Deep Reinforcement Learning," in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4005-4018, June 2019, doi: 10.1109/JIOT.2018.2876279

[294] Xiao, S., He, D., & Gong, Z. (2018, August). Deep-q: Traffic-driven qos inference using deep generative network. In *Proceedings of the 2018 Workshop on Network Meets AI & ML* (pp. 67-73).

[295] Xin He, Kaiyong Zhao, Xiaowen Chu, AutoML: A survey of the state-of-the-art, Knowledge-Based Systems, Volume 212, 2021, 106622, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2020.106622.

[296] Xiong, Z., Zhang, Y., Niyato, D., Deng, R., Wang, P., & Wang, L. C. (2019). Deep reinforcement learning for mobile 5g and beyond: Fundamentals, applications, and challenges. *IEEE Vehicular Technology Magazine*, *14*(2), 44-52.

[297] Xiong, Zehui, et al. "Deep reinforcement learning for mobile 5G and beyond: Fundamentals, applications, and challenges." *IEEE Vehicular Technology Magazine* 14.2 (2019): 44-52.

[298] Y. Chauvin, D. E. Rumelhart, “Backpropagation: theory, architectures, and applications”, Psychology press, 2013.

[299] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu and H. Zhang, "Deep Learning with Long Short-Term Memory for Time Series Prediction," in IEEE Communications Magazine, vol. 57, no. 6, pp. 114-119, June 2019, doi: 10.1109/MCOM.2019.1800155

[300] Y. Wang *et al*., "A Data-Driven Architecture for Personalized QoE Management in 5G Wireless Networks," in *IEEE Wireless Communications*, vol. 24, no. 1, pp. 102-110, February 2017. doi: 10.1109/MWC.2016.1500184WC.

[301] Y. Xu, W. Xu, Z. Wang, J. Lin and S. Cui, "Load Balancing for Ultradense Networks: A Deep Reinforcement Learning-Based Approach," in IEEE Internet of Things Journal, vol. 6, no. 6, pp. 9399-9412, Dec. 2019, doi: 10.1109/JIOT.2019.2935010.

[302] Yang, Jie, Shi Jin, Chao-Kai Wen, Jiajia Guo, and Michail Matthaiou. "3-D Positioning and Environment Mapping for mmWave Communication Systems." *arXiv preprint arXiv:1908.04142* (2019).

[303] Yanjiao Chen, Kishun Wu, and Qian Zhang, *From QoS to QoE: a tutorial on video quality assessment*, IEEE Communications Surveys & Tutorials

[304] Yu, H., Wu, Z., Wang, S., Wang, Y., & Ma, X. (2017). Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. Sensors, 17(7), 1501.

[305] Yu, Hwanjo, and Sungchul Kim. "SVM Tutorial-Classification, Regression and Ranking." Handbook of Natural computing 1 (2012): 479-506.

[306] Z. Li, M. Chen, K. Wang, C. Pan, N. Huang and Y. Hu, "Parallel Deep Reinforcement Learning Based Online User Association Optimization in Heterogeneous Networks," 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 2020, pp. 1-6, doi: 10.1109/ICCWorkshops49005.2020.9145209.

[307] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom and J. Frolik, "Non-Line-of-Sight Identification and Mitigation Using Received Signal Strength," IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, vol. 14, pp. 1689--1702, 2015.

[308] Z. Xu, Y. Wang, J. Tang, J. Wang and M. C. Gursoy, “A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs,” 2017 IEEE International Conference on Communications (ICC), Paris, 2017, pp. 1-6, doi: 10.1109/ICC.2017.7997286

[309] Zhai, Yanlong, et al. "Toward Reinforcement-Learning-Based Service Deployment of 5G Mobile Edge Computing with Request-Aware Scheduling." *IEEE Wireless Communications* 27.1 (2020): 84-91.

[310] Zhang, C. and Patras, P., 2018, June. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing* (pp. 231-240).

[311] Zhang, C., Ouyang, X. and Patras, P., 2017, November. ZipNet-GAN: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies* (pp. 363-375).

[312] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2019, May). Self-attention generative adversarial networks. In *International Conference on Machine Learning* (pp. 7354-7363). PMLR.

[313] Zhang, Shenglin, et al. "Prefix: Switch failure prediction in datacenter networks." Proceedings of the ACM on Measurement and Analysis of Computing Systems 2.1 (2018): 1-29

[314] Zhao, Zipiao, et al. "Service restoration in multi-modal optical transport networks with reinforcement learning." Optics Express 29.3 (2021): 3825-3840

[315] Zhou, H., Tan, L., Zeng, Q., & Wu, C. (2016). Traffic matrix estimation: A neural network approach with extended input and expectation maximization iteration. Journal of Network and Computer Applications, 60, 220-232.

[316] Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*, *107*(8), 1738-1762.

[317] Zhu, Mingyi, Kejiang Ye, and Cheng-Zhong Xu. "Network anomaly detection and identification based on deep learning methods." International Conference on Cloud Computing. Springer, Cham, 2018

[318] Project AFFORDABLE5G deliverable D1.2 “Affordable5G building blocks fitting in 5G system architecture”, https://www.affordable5g.eu/deliverables/

[319] 5G-SOLUTIONS deliverable D1.1A, “Definition and analysis of use cases/scenarios and corresponding KPIs based on LLs (v1.0)”, online: https://www.5gsolutionsproject.eu/wp-content/uploads/2020/03/D1.1A-Definition-and-analysis-of-use-cases-scenarios-and-corresponding-KPIs-based-on-LLs_FNET_v1.4_29102019-final2.pdf

[320] Project 5GVINNI Deliverable D1.4 “Design of infrastructure architecture and subsystems v2”, online: https://www.5g-vinni.eu/deliverables/

[321] Project 5GVINNI Deliverable D1.5 “5G-VINNI E2E Network Slice Implementation and Further Design Guidelines”, online: https://www.5g-vinni.eu/deliverables/

[322] 5G-SOLUTIONS deliverable D1.2A, “Cross-domain orchestration and management challenges analysis (v1.0)”, online: https://www.5gsolutionsproject.eu/wp-content/uploads/2020/03/Deliverable-D1.2A-Cross-domain-Service-Orchestration-and-Management-Challenges-Analysis-v1.0_31102019_FINAL-1.pdf

[323] G. Kakkavas, D. Gkatzioura, V. Karyotis, and S. Papavassiliou, “A Review of Advanced Algebraic Approaches Enabling Network Tomography for Future Network Infrastructures,” Future Internet, vol. 12, no. 2, p. 20, Jan. 2020 [Online]. Available: http://dx.doi.org/10.3390/fi12020020

[324] C. Chaieb, Z. Mlika, F. Abdelkefi and W. Ajib, "On the Optimization of User Association and Resource Allocation in HetNets With mm-Wave Base Stations," in IEEE Systems Journal, vol. 14, no. 3, pp. 3957-3967, Sept. 2020, doi: 10.1109/JSYST.2020.2984596.

[325] 5G-CLARITY, D4.1: "Initial Design of the SDN/NFV Platform and Identification of Target 5G-CLARITY ML Algorithms", October 2020

[326] 5G-CLARITY D4.2: "Validation of 5G-CLARITY SDN/NFV platform, interface design with 5G service platform, and initial definition of evaluation of ML algorithms", July 2021

[327] 5G-CLARITY D3.2: "Design refinements and initial evaluation of the coexistence, multi-connectivity, resource management and positioning frameworks", April 2021

[328] Friedman, J., Hastie, T. and Tibshirani, R., 2001. The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics.

[329] James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

[330] 5GZORRO consortium, "Deliverable 4.3. Final prototype of Zero Touch Service Mgmt with Security and Trust". Jan 2022.

[331] Corcoran, D., Kreuger, P., & Schulte, C. (2020, April). Efficient Real-Time Traffic Generation for 5G RAN. In NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium (pp. 1-9). IEEE.

[332] Breitgand, D., Lekidis, A., Behravesh, R., Weit, A., Giardina, P., Theodorou V., Costa, C. E., and Barabash, K. Dynamic Slice Scaling Mechanisms for 5G Multi-domain Environments. In 2021 European Conference on Networks and Communications (EuCNC). IEEE.

[333] Subramanya, T., & Riggio, R. (2021). Centralized and federated learning for predictive VNF autoscaling in multi-domain 5G networks and beyond. IEEE Transactions on Network and Service Management, 18(1), 63-78.

[334] Zhao, Z., Feng, C., Yang, H. H., & Luo, X. (2020). Federated-learning-enabled intelligent fog radio access networks: Fundamental theory, key techniques, and future trends. IEEE wireless communications, 27(2), 22-28.

[335] Theodorou, V., Lekidis, A. , Bozios, T., Meth, K., Fernández-Fernández, A., Taylor, J., Diogo, P., Martins, P., Behravesh, R. Blockchain-based Zero Touch Service Assurance in Cross-domain Network Slicing. In 2021 European Conference on Networks and Communications (EuCNC). IEEE.

[336] Behravesh, R., Perez-Ramirez, D. F., Rao, A., Harutyunyan, D., Riggio, R., & Steinert, R. (2020, November). ML-Driven DASH Content Pre-Fetching in MEC-Enabled Mobile Networks. In 2020 16th International Conference on Network and Service Management (CNSM) (pp. 1-7). IEEE.

[337] David Stephen Geoffrey Pollock, Richard C. Green and Truong Nguyen. Handbook of Time Series Analysis, Signal Processing, and Dynamics. 1999.

[338] Hochreiter S, Schmidhuber, Juergen. Long short-term memory. Neural computation. 1997;9(8):1735–80.

[339] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In NIPS 2014 Workshop on Deep Learning, December 2014.

[340] LOCUS – 5G PPP project, https://www.locus-project.eu/.

[341] Chafika Benzaid, Pol Alemany, Dhouha Ayed, G. Chollon, Maria Christopoulou, Gürkan Gür, … George. Xilouris. (2020, November 24). White Paper: Intelligent Security Architecture for 5G and Beyond Networks. Zenodo. http://doi.org/10.5281/zenodo.4288658

[342] 5G-CLARITY D3.1, 'State-of-the-Art Review and Initial Design of the Integrated 5G/Wi-Fi/LiFi Network Frameworks on Coexistence, Multi-Connectivity, Resource Management and Positioning,' August 2020.

List of editors/authors

Editors		
Daniel Camps Mur	i2CAT	Spain
Anastasius Gavras	Eurescom GmbH	Germany
Mir Ghoraishi	Gigasys Solutions	UK
Halid Hrasnica	Eurescom GmbH	Germany
Alexandros Kaloxylos	5G Infrastructure Association	Belgium
Authors / Contributors		
5G-COMPLETE		
Markos Anastasopoulos	National and Kapodistrian University of Athens	Greece
Anna Tzanakaki	National and Kapodistrian University of Athens	Greece
5G!Drones		
Gokul Krishna Srinivasan	Robots Expert Finland Oy	Finland
5Growth		
Kiril Antevski	Universidad Carlos III de Madrid	Spain
Jordi Baranda	Centre Tecnològic de Telecomunicacions de Catalunya	Spain
Koen De Schepper	NOKIA Bell	Belgium
Claudio Ettore Casetti	Politecnico di Torino	Italy
Carla Fabiana Chiasserini	Politecnico di Torino	Italy
Andres Garcia-Saavedra	NEC Laboratories Europe GmbH	Germany
Carlos Guimarães	Universidad Carlos III de Madrid	Spain
Koteswararao Kondepu	Scuola Superiore Sant'Anna and IIT Dharwad	Italy
Xi Li	NEC Laboratories Europe GmbH	Germany
Lina Magoula	National and Kapodistrian University of Athens	Greece
Marco Malinverno	Politecnico di Torino	Italy
Josep Mangues-Bafalluy	Centre Tecnològic de Telecomunicacions de Catalunya	Spain
Jorge Martín-Pérez	Universidad Carlos III de Madrid	Spain
Ricardo Martínez	Centre Tecnològic de Telecomunicacions de Catalunya	Spain
Chrysa Papagianni	NOKIA Bell	Belgium
Luca Valcarenghi	Scuola Superiore Sant'Anna	Italy
Engin Zeydan	Centre Tecnològic de Telecomunicacions de Catalunya	Spain
5GENESIS		

Özgü Alay	Simula Metropolitan Center for Digital Engineering	Norway
Erik Aumayr	Ericsson LMI	Ireland
Anne-Marie Cristina Bosneag	Ericsson LMI	Ireland
Giuseppe Caso	Simula Metropolitan Center for Digital Engineering	Norway
Elisa Jimeno	ATOS	Spain
5G-HEART		
Maria Diamanti	National Technical University of Athens	Greece
Grigorios Kakkavas	National Technical University of Athens	Greece
Vasileios Karyotis	National Technical University of Athens	Greece
Symeon Papavassiliou	National Technical University of Athens	Greece
Adamantia Stamou	National Technical University of Athens	Greece
5G-SOLUTIONS		
Anne-Marie Cristina Bosneag	Ericsson LMI	Ireland
Saman Feghhi	Ericsson LMI	Ireland
Ramy Mohamed	IBM	Ireland
Min Xie	Telenor	Norway
Sofiane Zemouri	IBM	Ireland
5G-TOURS		
Marco Gramaglia	University Carlos III of Madrid	Spain
5G-VINNI		
Thomas Dreibholz	Simula metropolitan center for digital engineering	Norway
Ahmed Elmokashfi	Simula metropolitan center for digital engineering	Norway
Sayantini Majumdar	Huawei Technologies	Germany
Joan Pujol Roig	Samsung Electronics R&D Institute	UK
Yue Wang	Samsung Electronics R&D Institute	UK
Wint Yi Poe	Huawei Technologies	Germany
Diego Lopez	Telefonica	Spain
Kashif Mahmood	Telenor	Norway
5GZORRO		
Rasoul Behravesh	Fondazione Bruno Kessler	Italy
Gino Carrozzo	NextWorks	Italy

Manuel Gil Pérez	University of Murcia	Spain
Alberto Huertas Celdrán	University of Murcia	Spain
José María Jorquera Valero	University of Murcia	Spain
Alexios Lekidis	Intracom Telecom	Greece
Gregorio Martínez Pérez	University of Murcia	Spain
Pedro Miguel Sánchez	University of Murcia	Spain
Tejas Subramanya	Fondazione Bruno Kessler	Italy
AFFORDABLE5G		
Josep Martrat	Atos Spain	Spain
Panagiotis Trakadas	National and Kapodistrian University of Athens	Greece
Anastasios Giannopoulos	National and Kapodistrian University of Athens	Greece
Sotirios Spantideas	National and Kapodistrian University of Athens	Greece
ARIADNE		
Edwin Yaqub	RapidMiner	Germany
Rachana Desai	RapidMiner	Germany
Ralf Klinkenberg	RapidMiner	Germany
Alexandros-Apostolos A. Boulogeorgos	University of Piraeus	Greece
Angeliki Alexiou	University of Piraeus	Greece
Jiguang He	University of Oulu	Finland
Nikos Katzouris	National Centre for Scientific Research Demokritos	Greece
Fotis Lazarakis	National Centre for Scientific Research Demokritos	Greece
Marco Di Renzo	Centre National de la Recherche Scientifique	France
Joonas Kokkoniemi	University of Oulu	Finland
INSPIRE-5Gplus		
Pascal Bisson	Thales	France
Dhouha Ayed	Thales	France
Geoffroy Chollon	Thales	France
Antonio Skarmeta	University of Murcia	Spain
Gür Gürkan	Zurich University of Applied Sciences	Switzerland
Chafika Benzaid	Aalto University	Finland
Edgardo Montes De Oca	Montimage	France

Antonio Agustin Pastor Perales	Telefonica	Spain
Internet of Radio Light		
John Cosmas	Brunel University	UK
Ben Meunier	Brunel University	UK
Lina Shi	Institut supérieur d'électronique de Paris	France
Xun Zhang	Institut supérieur d'électronique de Paris	France
LOCUS		
Andrea Conti	Consorzio Nazionale Interuniversitario per le Telecomunicazioni	Italy
Takai Eddine Kennouche	VIAVI Solutions	France
Flavio Morselli	Consorzio Nazionale Interuniversitario per le Telecomunicazioni	Italy
Chris Murphy	VIAVI Solutions	UK
MonB5G		
Hatim Chergui	Centre Tecnològic de Telecomunicacions de Catalunya	Spain
Francesco Devoti	NEC Laboratories Europe GmbH	Germany
Zhao Xu	NEC Laboratories Europe GmbH	Germany
Lanfranco Zanzi	NEC Laboratories Europe GmbH	Germany
SLICENET		
Jose Alcaraz Calero	University of the West of Scotland	UK
Qi Wang	University of the West of Scotland	UK
TERAWAY		
MohammedA Abdulkadir	Cumucore	Finland
Saimanoj Katta	Cumucore	Finland
Jose Costa-Requena	Aalto University	Finland
5G-CLARITY		
Anil Yesilkaya	University of Strathclyde	UK
Stefan Videv	University of Strathclyde	UK
Jonathan Prados-Garzon	University of Granada	Spain
Lorena Chinchilla-Romero	University of Granada	Spain
Pablo Muñoz-Luengo	University of Granada	Spain
Juan J. Ramos-Munoz	University of Granada	Spain

Carlos Colman-Meixner	University of Bristol	UK
Xueqing Zhou	University of Bristol	UK
Shuangyi Yan	University of Bristol	UK
Erik Aumayr	Ericsson LMI	Ireland
Meysam Goodarzi	IHP – Leibniz-Institut für innovative Mikroelektronik, Humboldt University of Berlin	Germany
Tezcan Cogalan	Interdigital	UK