



## Review

# From 4G to 5G: Self-organized network management meets machine learning

Jessica Moysen<sup>\*,a</sup>, Lorenza Giupponi<sup>b</sup>

<sup>a</sup> Department of Signal and Theory Communications, Universitat Politècnica de Catalunya (UPC), C/ Esteve Terradas, 7, Castelldefels (Barcelona) 08860, Spain

<sup>b</sup> Communications Network Division, Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Av. Carl Friedrich Gauss 7, Castelldefels (Barcelona) 08860, Spain

## ARTICLE INFO

## Keywords:

Network management  
Self-organizing networks  
Mobile networks  
Machine learning  
Big data

## ABSTRACT

Self-organization as applied to cellular networks is usually referred to Selforganizing Networks (SONs), and it is a key driver for improving Operations, Administration, and Management (OAM) activities. SON aims at reducing the cost of installation and management of 4G and future 5G networks, by simplifying operational tasks through the capability to configure, optimize and heal itself. To satisfy 5G network management requirements, this autonomous management vision has to be extended to the end to end network. In literature and also in some instances of products available in the market, Machine Learning (ML) has been identified as the key tool to implement autonomous adaptability and take advantage of experience when making decisions. In this paper, we survey how 5G network management, with an end-to-end perspective of the network, can significantly benefit from ML solutions. We review and provide the basic concepts and taxonomy for SON, network management and ML. We analyze the available state of the art in the literature, standardization, and in the market. We pay special attention to 3rd Generation Partnership Project (3GPP) evolution in the area of network management and to the data that can be extracted from 3GPP networks, in order to gain knowledge and experience in how the network is working, and improve network performance in a proactive way. Finally, we go through the main challenges associated with this line of research, in both 4G and in what 5G is getting designed, while identifying new directions for research.

## 1. Introduction

Traditionally, and up to 4G, the evolution from one generation of mobile networks to another, has been driven by hardware technology advancements. The revolution of 5G is different, and novel advancements of software technology will be critical, especially in the way the network will be managed. With the advent of these software advancements, and unprecedented levels of computational capacity, the vision of autonomous Network Management (NM) can be put into practice taking advantage of cross-disciplinary knowledge advancements in the area of ML. This vision aligns with the concepts of self-awareness, self-configuration, self-optimization, and self-healing, which have already been defined in the area of NM, in the access segment of Long Term Evolution (LTE), through the concept of Self-organizing Network (SON).

SON is a common term used to refer to mobile network automation and minimization of human intervention in the cellular/wireless network management. This concept has been introduced by 3GPP in Release 8 and it has been expanding across subsequent releases. 3GPP

work is inspired by a set of requirements defined by the operators' Alliance Next Generation Mobile Networks (NGMN). The main objective of SON can be roughly classified into three main points: 1) to bring intelligence and autonomous adaptability into cellular networks, 2) to reduce capital and operational expenditures (CAPEX/OPEX), 3) to enhance network performances in terms of network capacity, coverage, spectral efficiency, offered service/experience, etc. In particular, SON is a driving technology providing capabilities for *self-configuration* to automatize the network deployment phase, *self-optimization* to optimize the network performance while it is operating, by properly adjusting network parameters, and *self-healing* to detect faults and to correct/prevent network failures. The overall complex SON problem has been decomposed in a set of useful use cases, which have been identified by 3GPP, NGMN, 5G Infrastructure Public Private Partnership (5GPPP) and different EU projects [1–6]. The academic literature has dedicated significant effort to SON algorithms in the context of the above mentioned 4G use cases, providing smart solutions to optimize network operator performance, expenses and users' experience. Many of these works are already reviewed here [7]. The market also offers already

\* Corresponding author.

E-mail addresses: [jessica.moysen@tsc.upc.edu](mailto:jessica.moysen@tsc.upc.edu) (J. Moysen), [lorenza.giupponi@cttc.es](mailto:lorenza.giupponi@cttc.es) (L. Giupponi).

<https://doi.org/10.1016/j.comcom.2018.07.015>

Received 13 January 2018; Received in revised form 4 July 2018; Accepted 6 July 2018

Available online 27 July 2018

0140-3664/ © 2018 Elsevier B.V. All rights reserved.

complete sets of SON solutions, (e.g. [8–12], among others) many products have been advertised and presented in Mobile World Congress (MWC) 2016–2018 [13–16]. For example, AirHop's eSON from Jio & AirHop communications [11], which employs a multi-vendor, multi-technology, real-time SON solution based on scalable and virtualized software platform has recently been awarded for the 2016 Small Cell Forum Heterogenous Network (Het-Net) management software and service award [17].

Up to date SON solutions have been designed as an add-on feature to LTE, associated to a specific set of use cases affecting the Radio Access Network (RAN). As a result, as of today, SON represents mainly a RAN solution, rather than an integral feature of the end-to-end network, capable of fulfilling the requirements of different kinds of vertical service instances [18]. In addition, several open challenges are still unsolved, like the problem of coordination of SON functions [19,20], or the proper solution of the trade-off between centralized and distributed SON implementations [21]. To achieve this vision, the networking world is exploring new directions. On the one hand, Software Defined Network (SDN) is expected to bring the benefits of programmability and flexibility to the management of the network, while Network Functions Virtualisation (NFV) is expected to extend the economy of scale of the Information Technology industry to the Telecom industry, by simplifying the deployment of new features and functionalities through implementation in Commercial off-the-shelf (COTS) without the need of specialized hardware. These virtualized functions can be centralized in the cloud (e.g. cloud-RAN) or can be migrated to the edge through Mobile Edge Computing (MEC), where new processing capabilities are introduced in the base station for new applications [22]. On the other hand, as we observed in [23] a huge amount of data is currently already generated in 4G networks during normal operations by control and management functions, and more is expected to come in 5G networks due to the densification process [24], heterogeneity in layers and technologies, the additional control and management complexity in NFV and SDN architectures, the advent of Machine to Machine (M2M) and Internet of Things (IoT) paradigms, the increasing variety of application and services, each with distinct traffic patterns and Quality of Service (QoS)/Quality of Experience (QoE) requirements, etc. As a result, 5G architectures should support Big Data platforms to handle measurements, traces, logs and configuration data from the entire network, providing the basis for automatic decision making.

5G network management is expected to provide a whole new set of challenges due to 1) the need to manage future network complexity, associated to ultra dense deployments, heterogeneous nodes, networks, applications, different RANs coexisting in the same setting, 2) the need to manage very dynamic networks, where operators may not have complete control on the deployment of some nodes (e.g. femto-cells deployed by end users), energy saving policies are in place generating a fluctuating number of nodes, active antennas are a reality, etc. 3) the need to support 1000x traffic, and 10x users, and improve energy efficiency, 4) the need to improve the experience of the users by enabling Gbps speeds, and highly reduced latency, 5) the need to manage new virtualized architectures, 6) the need to handle heterogeneous spectrum access privileges through novel LTE-Unlicensed (LTE-U), Licensed Assisted Access (LAA), MuLTEFire, New Radio (NR), NR-U paradigms and the availability of both traditional sub- 6 GHz bands, and above 6 GHz mmWave bands. In this challenging context, we believe that the use of next generation (NG) SON and of smart network management policies is crucial and inevitable for operators running multi-RAT, multi-vendor, multi-layer networks, where an overwhelming number of parameters need to be configured and optimized.

We believe that Machine Learning (ML) can be effectively used to allow the network to learn from experience, while improving performance. In particular, big data analytics, through the analysis of data already generated by the network, can pursue the self-awareness by driving the network management from reactive to predictive. The use of NG SON based on ML allows operators to make the networks 1) more

self-aware, by exploiting the information already available in the network to gain experience in the network management, 2) more self-adaptive, by exploiting intelligent control decisions procedures which allow to automatize the decision processes based on the experience. In this paper, we focus on the application of ML to 5G network management. Differently from other surveys on SON proposed before [7,25,26] or from other surveys related with 5G network management [27], we focus here on the study and analysis of the available literature on SON and network management considering ML as the tool to implement automation and self-organization, from a 5G perspective. Another work [26] has focused on the application of ML to SON, with emphasis on 4G LTE and on the specific set of SON use cases defined for the LTE RAN. However, differently from this work, in this paper we focus on NG SON for 5G networks, and on how ML will be the driving tool for autonomous end-to-end 5G network management. In particular, we review and provide the basic concepts and taxonomy of traditional SON and 5G network management in Section 2. We pay special attention to the evolution of 3GPP in the area, following its nomenclature. We then provide, in Section 3, guidelines to select the most appropriate ML algorithm and approach, based on the network management issue to address. In Section 4, we review the main sources of information relevant for a knowledge based network management, data that is actually already generated by the network, and we survey the literature on ML-based network management. We then highlight challenges for future works in Section 5. Finally, Section 6 concludes the survey.

## 2. Self organizing network (SON) and network management

SON is a key driver to maximize total performance in cellular networks. The main idea is to bring into them intelligence and autonomous adaptability by diminishing human involvement, while enhancing network performance, in terms of network capacity, coverage and service quality.

The main motivation behind the increasing interest in the introduction of SON from operators, standardization bodies and projects is twofold. On the one hand, from the market perspective, the ever increasing demand for a diversity of offered services, and the need to reduce the time to market, further add to the pressure to remain competitive by effectuating cost reductions. On the other hand, from the technical perspective, the complexity and large scale of future radio access technologies imposes significant operational challenges due to the multitude of tunable parameters and the intricate dependencies among them. In addition, the advent of heterogeneous networks is expected to tremendously increase the number of nodes in this new ecosystem, so classic manual and field trial design approaches are just impractical.

Similarly, manual optimization processes or fault diagnosis and cure, performed by experts are no longer efficient and need to be automatized, as this causes time intensive experiments with limited operational scope, or delayed, manual and poor handling of cell/sites failures. Key operational tasks, such as radio network planning and optimization are largely separated nowadays and this causes intrinsic shortcomings like the abstraction of access technologies for network planning purposes, or the consideration of performance indicators that are of limited relevance to the end user's service perception. These problems have been approached through SON by European projects such as SOCRATES [2], and Gandalf [1]. Also FP7 and 5GPPP EU projects have been dealing with SON. In particular FP7 SEMAFOR [3], which develops a unified self-management to operate complex HetNets. Among 5GPPP projects, we highlight SESAME [4], which proposes the Cloud-Enabled Small Cell (CESC) concept, i.e., a new multi-operator enabled Small Cell by deploying Network Functions Virtualisation (NFV), supporting powerful self-management inside the access network infrastructure. In terms of self-organized network management in SDN and NFV, the SELFNET project aims at enabling the use of ML to achieve real time autonomous 5G network management [5]. In

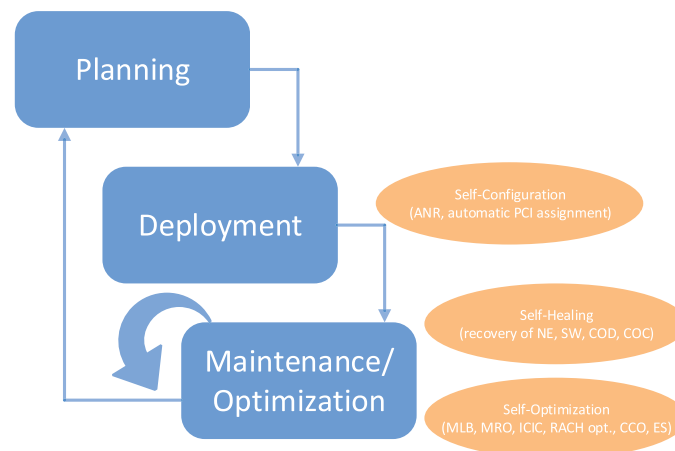


Fig. 1. Self-organizing networks.

particular, the project explores a smart integration of state-of-the-art technologies in SDN, NFV), SON, Cloud computing, Artificial intelligence. The COGNET project [6] has similar objectives and aims at developing several operators' use cases by applying ML algorithms.

SON has been introduced by 3GPP as a key component of LTE network starting from the first release of this technology in Release 8, and expanding to subsequent releases. In SOCRATES [2] and in 3GPP [28], meaningful SON use cases have been defined, which can be classified according to the phases of the life cycle of a cellular systems (planning, deployment, maintenance and optimization) into: self-configuration, self-healing and self-optimization, as depicted in Fig. 1. In this section, first we give an overview of the evolution of SON in 3GPP. We go through self-configuration, optimization and healing functionalities, introducing the use cases that have been defined for each one of them. We discuss about the self-coordination problem, to handle the potential conflicts that may exist between the parallel execution of multiple SON functions. We then present a standardized database used for different 3GPP use cases known as the Minimization of Drive Tests (MDT) functionality. Finally, we focus on an end-to-end vision by extending SON principles to the core, and we discuss the role of virtualized and software defined architectures in the context of 5G Network Management. Notice that here we do not focus on the academic literature, as it has already been reviewed in other interesting works [7,26]. Differently from previous work, we focus on the taxonomy defined by 3GPP, on the related roadmap, and we pay attention to the market penetration.

### 2.1. SON evolution in 3GPP

3GPP Release 8 started defining LTE and already sets the basis for concepts and requirements, and for SON functionalities regarding self-configuration, initial equipment installation and integration. The Automatic Neighbour Relation (ANR) functionality is introduced here to reduce manual work when configuring the neighbouring list in newly deployed eNBs. Concepts of self-optimization are defined in the context of Release 9. It includes optimization of coverage, capacity, handover and interference. The functions which are introduced (and that will be detailed in the following sections) are Mobility Load Balancing (MLB), Mobility Robustness/Handover Optimisation (MRO), Inter-Cell Interference Coordination (ICIC) and Random Access Channel (RACH) optimization. Release 10 focuses on improvements to already defined SON functions to enhance interoperability between small cells and macro-cells and includes NGMNs recommendations, i.e., new functionalities such as Coverage and Capacity Optimization (CCO), enhanced ICIC, and it defines all the concepts related to self-healing, so Cell Outage Detection (COD) and Cell Outage Compensation (COC)

functions. Finally, concepts of MDT and Energy Saving (ES) are also introduced and then enrich in Release 11. Release 11 SON functions are related to the automated management of heterogeneous networks. It includes mobility robustness optimization enhancements and inter-radio access technology Handover (HO) optimization. Release 12 introduces optimization and enhancements for small cells including deployments in dense areas. In Release 13 novel concepts of unlicensed LTE have been introduced. Besides that, Release 13 studied the intensification of OAM, with respect to centralized and distributed architecture. In particular, focuses on distributed MLB, as well as on enhanced NM or centralized CCO. Release 14 focuses on meeting the 5G requirements in terms of latency reduction, use of unlicensed spectrum in a fair manner, support for carrier aggregation, energy efficiency at OAM level, SON for active antennas, etc. Finally, Release 15 has studied the pre-configuration of the eNB with alternative coverage configurations to select the coverage change during Cell Shaping and set the proper handover margin with neighbour cells. Self-Configuration and Self-Optimization are not finalized yet, and are targeted for completion in June 2018 [82]. Other documents of interest also include the protocol neutral SON policy Network Resource Model (NRM) Integration Reference Point (IRP), with the Information Service (IS) [37,47] and Solution Sets (SS) [48,49]. Table 1 summarizes the evolution of SON in 3GPP.

### 2.2. SON implementations

In this section, we discuss about the different architectures that have been considered for SON, ranging from Centralized SON (C-SON) functionalities, where the self-organizing algorithms reside in the network management system, in the OAM or in the Network Management Systems (NMS), to a Distributed SON (D-SON) solution, where the SON functions are distributed, in the control plane, across the edges of the network, typically in the Enhanced Node Base stations (eNBs). On the one hand, C-SON can take into consideration data from all nodes in the network to identify and address network-wide issues. It can potentially orchestrate the behavior of radio network equipments across an entire network of multi-vendor and multi-technology environments. C-SON solutions can be more robust against network instabilities caused by the concurrent operation of SON functions with conflicting objectives because control is centralized. However, centralized systems may respond too slowly in the emerging world of small cells that experience very transitory traffic loads. The SON implementations are depicted in Fig. 2.

Market implementations of C-SON are offered by vendors like Celcote (acquired by AMDOCS), Ingenia Telecom and Intucell (acquired by Cisco), while D-SON solutions have traditionally been more challenging to implement and vendor specific, not allowing for easy

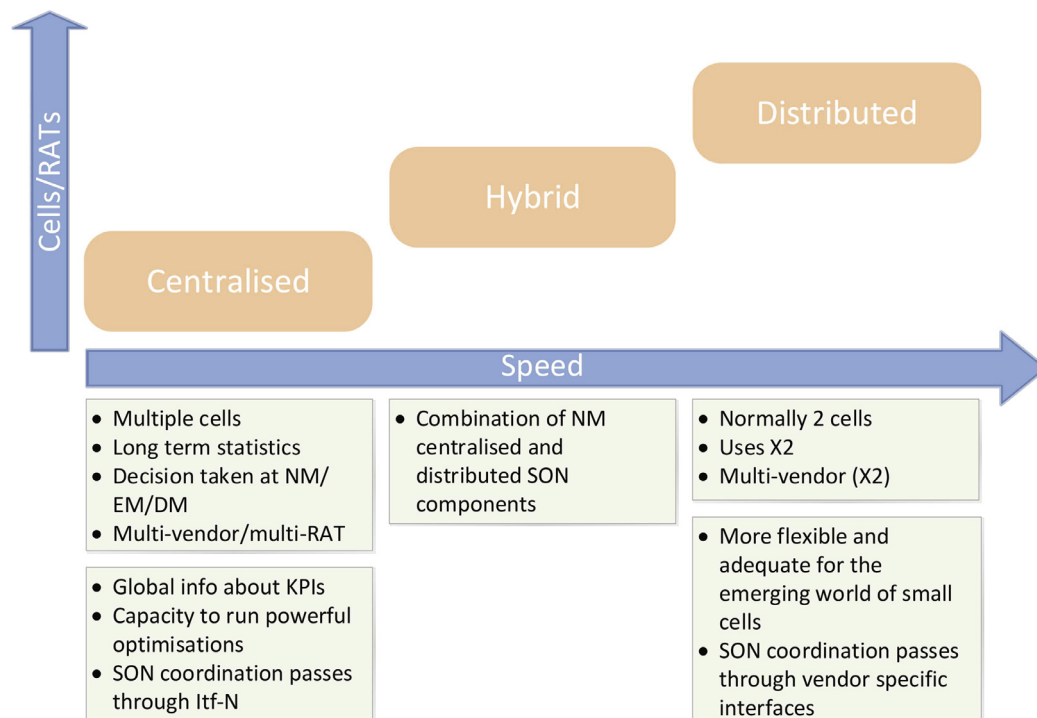
**Table 1**  
Evolution of SON in 3GPP.

Release	WI	Feature	TS or TR
Rel.8	SA5-SON concepts and requirements	SON concepts and requirements	[29]
Rel.8	SA5-Self establishment of eNBs	Self configuration	[30–35]
Rel.8	SA5-SON Automatic Neighbour Relation (ANR) list management	ANR, PCI	[36–39]
Rel.9	SA5: Study of SON related OAM Interfaces for HeNBs	SON related OAM Interfaces for HeNBs	[40]
Rel.9	SA5: Study of self-healing of SON	Self-healing management	[41]
Rel.9	SA5:SON OAM aspects: Automatic radio network configuration data preparation	Automatic radio network configuration data preparation	[30–32]
Rel.9	SA5:SON OAM aspects self-organization management	Self-optimization (MRO, MLB, ICIC)	[42]
Rel.9	RAN3: Self-organizing networks	CCO, MRO, MLB, RACH opt.	[43–46]
Rel.10	SA5: SON self-optimization management continuation	Self-coordination, self-optimization (MRO, MLB, ICIC, RACH opt.)	[37,42,47–49]
Rel.10	SA5: Self-healing management	CCO, COC	[50]
Rel.10	SA5: OAM aspects of ES in radio networks	ES	[37–39,42,51–53]
Rel.10	RAN2-3: LTE SON enhancements	CCO, ES, MLB, MRO enhancements	[44–46,54]
Rel.11	SA5: ULTRAN SON management	SON management	[29,47,48,55–58]
Rel.11	SA5: LTE SON coordination management	SON coordination [59]	[29,37,42,47,48,57]
Rel.11	SA5: Inter RAT ES management	OAM aspects of ES management	[47,48,53,55,58,60]
Rel.11	RAN3: Further SON enhancements	MRO, MDT enhancements	[43–46,54,61–64]
Rel.12	SA5: Enhanced NM centralized CCO	Enhanced NM centralized CCO	[42,65–70]
Rel.12	SA5: Multi-vendor plug and play eNB connection to the network	Multi-vendor plug and play eNB connection to the network	[30,71,72]
Rel.12	SA5: Enhancements on OAM aspects of distributed MLB	OAM aspects of distributed MLB	[73]
Rel.12	SA5: Energy efficiency related performance measurements	Energy efficiency related performance measurements	[42]
Rel.12	SA5: Het-Nets management/OAM aspects of network sharing	Het-Nets/network sharing	[74,75]
Rel.12	RAN2-3: Next generation SON for ULTRAN/EUTRAN	SON per UE type, active antennas, small cells	[76]
Rel.12	RAN2-3: ES enhancements for EUTRAN	ES	[77]
Rel.13	RAN2-3: Enhanced Network Management centralized CCO	CCO	[67]
Rel.13	SA5: Study on Enhancements of OAM aspects of Distributed Mobility Load Balancing SON function	MLB	[78]
Rel.14	RAN: OAM (SON for Active Antenna Systems (AAS)-based deployments)	Energy efficiency	[79,80]
Rel.15	RAN: OAM (Study on OAM aspects of SON for AAS-based deployments)	Coverage configuration	[81]

interaction of products from different vendors, so that a supervisory layer is commonly still needed to coordinate the different instances of D-SON across a much broader scope and scale. Only recently, vendors like Qualcomm or Airhop have started proposing D-SON as a SON mainstream, as small cells and Het-Net require the millisecond response times of D-SON.

### 2.3. Self configuration

Self-configuration is the process of bringing a new network element into service with minimal human operator intervention [30]. This covers the cellular system life cycle phase related to planning and deployment. Self-configuring algorithms take care of all configuration aspects of the eNB. When the eNB is powered on, it detects the transport



**Fig. 2.** SON implementations.



link and establishes a connection with the core network elements. After this, the eNB is ready to establish OAM, S1 and X2 links and finally sets itself in operational mode. After the eNB is configured, it performs a self-test to deliver a status report to the network management node. Since Release 8 ANR and Automated Configuration of Physical Cell Identity (PCI) use cases have been considered [83,84]. The ANR function resides in the eNB and manages the conceptual Neighbour Relation Table (NRT). Located within ANR, the Neighbour Detection Function finds new neighbours and adds them to the NRT. ANR also contains the Neighbour Removal Function which removes outdated NRs. The Neighbour Detection Function and the Neighbour Removal Function are implementation specific [85]. The PCI is a physical layer signature to distinguish signals from different eNBs. It is based on synchronization signals. The total number of PCIs in LTE is 504, so that reuse is inevitable, especially in dense deployments. The Automatic PCI assignment aims at an automatic conflict and confusion free identification of cells [86]. Recommended practices for both use cases can be found in [87].

#### 2.4. Self optimization

Self-optimization embraces all the set of mechanisms which optimize the network parameters during operation, based on measurements received from the network. In the following we provide a brief overview of the main self-optimization function that have been introduced across the different recent releases [86]. From Release 9, we highlight work on:

1. **MLB.** The MLB is the SON function in charge of managing cells' congestion through load transfer to other cells. The main objective is to improve the end-user experience and achieve higher system capacity by distributing user traffic across the system radio resources. The implementation of this function is generally distributed and supported by the load estimation and resource status exchange procedure. The messages containing useful information for this SON function (resource status request, response, failure and update) are transmitted over the X2 interface [46]. MLB can be implemented by tuning the Cell Individual Offset (CIO) parameter. The CIO contains the offsets of the serving and the neighbour cells that all UEs in this cell must apply in order to satisfy the A3 handover condition [88].
2. **MRO.** The MRO is a SON function designed to guarantee proper mobility, i.e. proper handover in connected mode and cell re-selection in idle mode. Among the specific goals of this function we have the minimization of call drops, the reduction of Radio Link Failures (RLFs), the minimization of unnecessary handovers, ping-pongs, due to poor handover parameters settings, the minimization of idle problems. Its implementation is commonly distributed. The messages containing useful information are: the S1AP handover request or X2AP handover request, the handover report, the RLF indication/report. Release 11 focused on different improvements of the handover optimization [89]. MRO operates over connected mode and idle mode parameters. In connected mode, it tunes meaningful handover trigger parameters, such as the event A3 offset (when referring to intra-RAT, intra-carrier hand-overs), the Time to Trigger (TTT), or the Layer 1 and Layer 3 filter coefficients. In idle mode, it tunes the offset values, such as the Qoff-set for the intra-RAT, intra-carrier case.
3. **Inter-Cell Interference Coordination (ICIC).** ICIC aims to minimize interference among cells using the same spectrum. It involves the coordination of physical resources between neighbouring cells to reduce interference from one cell to another. ICIC can be done in both uplink and downlink for the data channels Physical Downlink Shared Channel (PDSCH), and Physical Uplink Shared Channel (PUSCH), or uplink control channel Physical Downlink Control Channel (PDCCH). ICIC can be static, semi-static or dynamic. Dynamic ICIC relies on frequent adjustments of parameters,

supported by signalling among cells over X2 interface. To support proactive coordination among cells the High Interference Indicator (HII) and the Relative Narrowband Transmit Power (RNTP) indicators have been defined, while to support reactive coordination, the Overload Indicator (OI) has been introduced [46].

4. **RACH.** RACH optimization aims at optimizing the random access channels in the cells based on UE feedback and knowledge of its neighbouring eNBs RACH configuration. RACH optimization can be done by adjusting the Power control (Pc) parameter or change the preamble format to reach the set target access delay [90].

In Release 10, 3GPP defined new use cases.

1. **Coverage and Capacity Optimization (CCO)** is a SON function that aims to design self-optimizing algorithms that achieve optimal trade-offs between coverage and capacity. Different mechanisms can be considered to dynamically improve coverage and capacity, such as ICIC, scheduling, and the combination of such mechanisms. The targets that can be optimized may be vendor dependent and include coverage, cell throughput, edge cell throughput, or a weighted combination of the above.
2. **ES** aims at providing the quality of experience to end users with minimal impact on the environment. The objective is to optimize the energy consumption, by designing Network Elements (NEs) with lower power consumption and temporarily shutting down unused capacity or nodes when not needed [51]. In particular, many works in literature have been focusing on switching ON/OFF eNBs or small cells, in an efficient way, in order to guarantee a target level of Quality of service/experience, while minimizing the dissipated energy.

Release 11 provides enhancements to MLB optimization, HO optimization, CCO, and ES. Release 12 has focused on a study on enhancements of OAM aspects for distributed MLB [78].

#### 2.5. Self-healing

Self-healing [50] focuses on the maintenance phase of a cellular network. Wireless cellular systems are prone to faults and failures, and the most critical domain for fault management is the RAN. Every eNB is responsible for serving an area, with little or none redundancy. If a NE is not able to fulfill its responsibilities, it results in a period of degradation of performances, during which users are not receiving a proper service. This results in severe revenue loss for the operator. Self-healing was initially studied in Release 9 [41], but it is in Release 10, when the main work has been carried out and features for detection, and adjustment of parameters have been specified [91]. These specifications have been further updated in Release 11 [50]. The main defined use cases are the following.

1. **Self-recovery of NE Software.** If the NE software failed due to load earlier software version and/or configuration, the most important thing is to ensure that the NE runs normally by removing the fault software, and restoring the configuration.
2. **Self Healing of board Faults.** This use case aims to solve hardware failures in the NE [92].
3. **Cell Outage Management.** This use case is split in two main functions: 1) Cell Outage Detection. The main objective here is to detect a cell outage through the monitor performance indicators, which are compared against thresholds and profiles, and 2) Cell Outage Compensation. This use case aims at alleviating the outage caused by the loss of a cell from service [50]. It refers to the automatic mitigation of the degradation effect of the outage by appropriately adjusting suitable radio parameters, such as the pilot power and the antenna parameters of the surrounding cells.

## 2.6. Self coordination

SON functionalities are often designed as stand-alone functionalities, by means of control loops. When they are executed concurrently in the same or different network elements, the impact of their interactions is not easy to be predicted, and unwanted effects may even occur among instances of the same SON function, when implemented in neighbouring cells. The risk of unacceptable oscillations of configuration parameters or undesirable performance results increase with the number of SON functions.

As we stated earlier, 3GPP has proposed different architectures for SON implementation, ranging from centralized C-SON to distributed D-SON. The choice of the architecture has a strong impact on the efficiency of the self-coordination framework. If C-SON is used, SON functions are implemented in the OAM or in the NMS, as part of the Operation and Support System (OSS). This implementation benefits from global information about metrics and Key Performance Indicator (KPI)s, as well as computational capacity to run powerful optimization algorithms involving multiple variables or cells. However, it suffers from long time scales. In order to avoid oscillations of decision parameters, 3GPP requires [59] that each SON function asks for permission before changing any configuration parameter. This means that a request must be sent from the SON function to the SON coordinator and a response has to be returned. In C-SON all these requests must pass through the Interface-N, which is not suitable for real-time communication, so that there is no possibility to give priority to SON coordination messages over other OAM messages. If in turn, distributed coordination is used, the interaction between the SON function and the local SON coordinator will be over internal vendor-specific interfaces, with much lower latency characteristics. This makes the D-SON architecture much more flexible and adequate for small cell networks, which experience very transitory traffic loads, thus requiring high reactivity to propagation and traffic conditions.

An example of this can be observed in Fig. 3, where we provide an analysis of how the iterations among several SON functions implemented in centralized and distributed manner can generate conflicts in the network. In particular, this figure focuses on the SON output parameter conflict, i.e., when two or more SON functions aim at optimizing the same output parameter with different actions requested, and where at least three possible conflicts can arise: 1) the resource conflict between MRO and MLB, which is generated by the concurrent execution of actions 3 and 4; 2) the one among CCO and ICIC, which is generated by the concurrent execution of actions 1 and 6, and/or 3) the one among COC and ICIC, which is generated by the concurrent execution of actions 2, 5 and 7. We can identify output parameters, which are affected by two opposite decisions of two different functions, trying to achieve their own targets. As a result, to define and implement a self-coordination framework is considered a necessity [2,93,94].

## 2.7. Minimization of drive tests

MDT enables operators to collect User Equipments (UEs) measurements together with location information, if available, with the purpose of optimizing network management while reducing operational effects and maintenance costs. This feature has been studied by 3GPP since Release 9 [95], among the targets there are the standardization of solutions for coverage optimization, mobility, capacity optimization, parametrization of common channels, and QoS verification [92]. Since operators are also interested in estimating QoS performance, in Release 11, MDT functionality has been enhanced through QoS performance to properly dimension and plan the network by collecting measurements indicating throughput and connectivity issues [96]. These MDT functions have been further elaborated in Release 11, while Release 12 has included specific enhancements in terms of correlation of information, which can be found in the study on enhanced network management

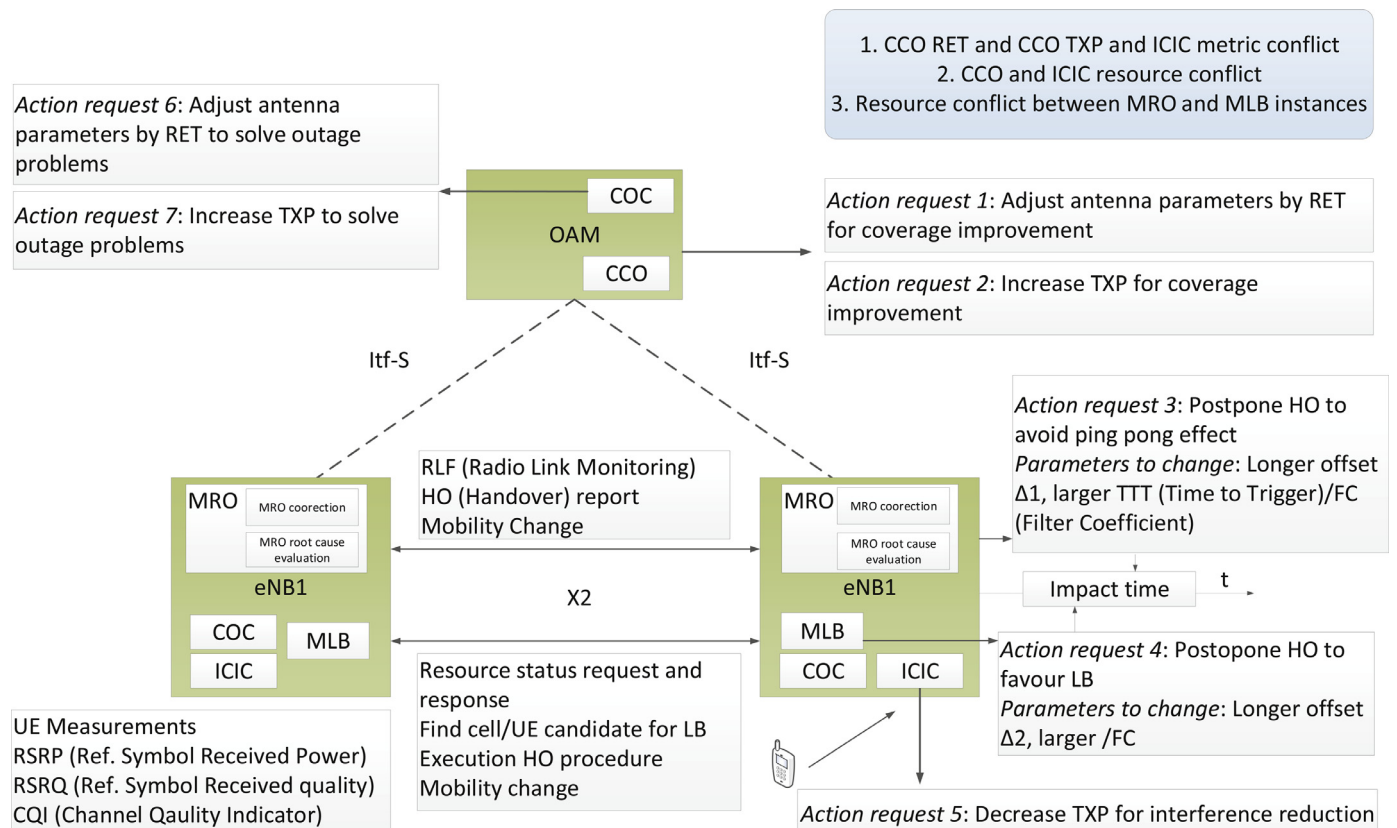


Fig. 3. High-level example of how the iterations of multiple SON functions may interfere.

centralized CCO. These improvements and extensions of SON enhancements introduced until Release 13 can be found in [29].

## 2.8. Core networks

The core network operations can be managed through self-organizing functionalities. The benefits also in this case come from the reduced human intervention and from reduced operational costs. self-organization in the core network allows to self-adapt traffic loads and prevent bottlenecks. In addition, self-organization for Core enables the core network to handle signalling more efficiently. In this regard, Nokia [97] already automates core networks operations based on SON technology. The objective is to automatically and rapidly allocate core network resources to meet unpredictable behaviours and demands in terms of broadband. Notice that SON use cases for core networks are not limited to LTE networks, but many of them can be taken into account also for other kinds of networks, like 2/3G.

## 2.9. SON in virtualized and softwarized 5G architecture

The wireless industry is currently working towards being prepared for a 1000x data traffic growth. It is unlikely, though, that users will want to pay more for the service than they are paying today, which set a serious challenge for both mobile operators and vendors, i.e. how to improve the infrastructure 1000 times, without increasing the CAPEX and OPEX. Besides SON, another trend in this direction, initiated by an European Telecommunications Standards Institute (ETSI) industrial study group in 2012, is the NFV, which allows to exploit the economies

of scale of the IT industry, by moving traditional network functions away from specialized hardware to general purpose computation, storage and memory pools, distributed throughout the network and in data centers. NFV virtualizes the functional elements of the network, instantiating the corresponding functions as programs that run on COTS, and less expensive hardware. This concept, combined with a SDN architecture, in charge of exposing the network resources, through Application Programming Interfaces (APIs), to external applications, is introduced to make mobile network deployments more cost-effective [27,98].

On the one hand, SDN is promoted by the Open Networking Foundation (ONF) and defines an architecture of three layers, the Data layer, the control layer and the application layer. The data layer exposes the hardware resources towards the control layer, which manages the network devices according to policies or high level applications. The application layer allows the development of high level applications to customize and program the network behaviour using well defined interfaces. Communication between the layers is handled by the Northbound and Southbound interfaces, as shown in Fig. 4(a).

On the other hand, the virtualization refers to the abstraction of logical from physical resources. In this context, NFV proposes to transfer network functions, such as routing, firewall, gateway, deep packet inspection, etc., towards virtual software-based applications, executed on Information Technology (IT) platforms (servers, switches, storage). This allows improved flexibility and scalability, which facilitates the implementation of new network functions, while decreasing costs. For this, the network functions have to migrate from being implemented in specialized hardware to run on top of COTS hardware.

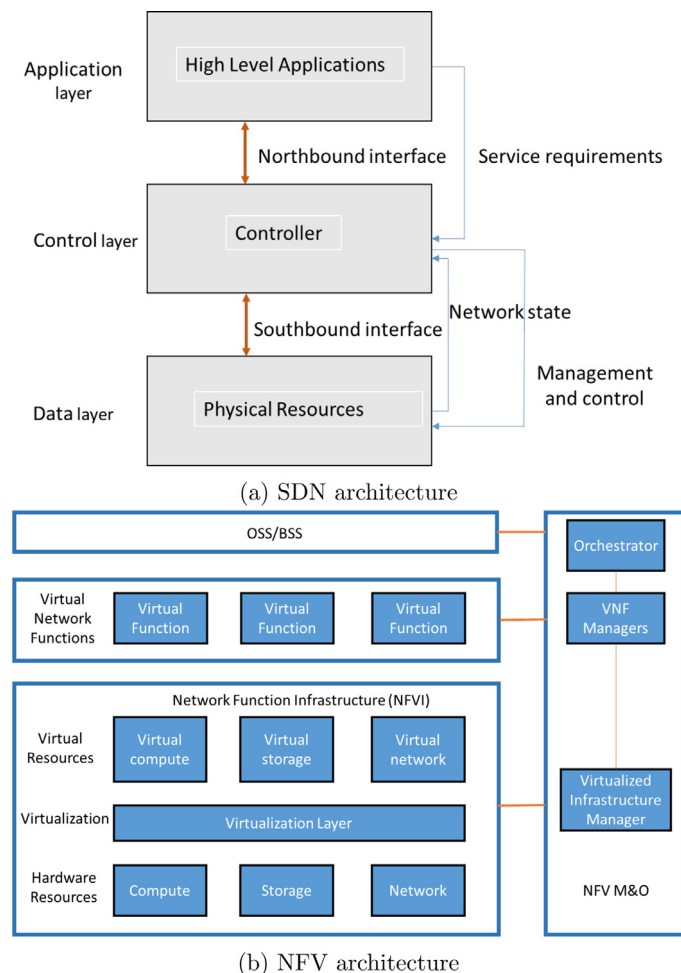


Fig. 4. SDN and NFV architectures.

The ETSI has proposed a NFV architecture based on three main modules, the NFV Infrastructure (NFVI), NFV and the NFV Management and Orchestration, as it is shown in 4. NFVI represents the hardware and software resources where NFV is applied. The virtualization layer allows the abstraction of the different resources. The Virtual Network Function (VNF) represents the virtual instance that runs over the NFVI. Finally, the orchestration and management of VNF and NFVI ensures the optimal operation of VNFs over the available infrastructure. The orchestrator, the VNF manager and the Virtualized Infrastructure Manager, in charge of the resource inventory to check on resource availability, ensures the proper intelligence of the network. This layer is the brain with the broadest view of the network, it coordinates the allocation of functions across the different segments of the dense, heterogeneous network. When the RAN is provided with IT and cloud computing capabilities by deploying COTS at the access side, we talk about MEC. MEC aims to place the service platform at the edge of the network, to provide nearby service computing and data caching for mobile users. It utilizes virtualization and cloud computing technologies to run and manage applications at the edge of the mobile network, to provide low latency and high bandwidth experience. Both MEC and NFV architectures require a network infrastructure virtualization platform, and an orchestrator making decisions to meet the requirements of the operational system and service applications. At the methodological level, the orchestrator can take advantage of the huge amount of information travelling through the network, in terms of measurements, signaling information, QoS and QoE indicators, etc., by means of machine learning based approaches.

Even if SDN and NFV have already a value when implemented independently, they have an added value when they are combined. The NFV technology allows to easily create and program new network functions and to place them in the most suitable location in the network, coping with capacity, coverage and quality issues, allowing seamless integration of new network elements, all while using the most appropriate amount of resources. This requires though the SDN to be able to adjust the network accordingly, enabling network programmability.

In this context, SON can be useful to achieve real time autonomous network management. In this novel softwarized visions, we can benefit of all the opportunities offered by centralized, distributed and local

implementations proposed for SON, by proposing a SON over NFV architecture, where SON functions, aimed at tackling the main radio access and backhauling challenges of extremely dense deployments, are virtualized and run over generic purpose hardware. These virtualized SON functions can be deployed in decentralized manner (D-SON), close to the edge of the network, following the MEC paradigm, or in centralized nodes (C-SON), in order to fulfill tradeoffs between latency and optimal operations, and while responding online to the specific situation of the network in terms of load, perceived QoE/QoS, etc. Moreover, SON operating over a SDN/NFV architecture would be compatible also with non-3GPP systems. The market offers start-ups which already advertise the concept of virtual-SON in the cloud. Cellwize [13] is one of them.

However, this research line is extremely novel. A joint standardized architecture including fundamental concepts like NFV and MEC, SDN, and SON is still under discussion and cannot be easily defined here. We want to highlight the work that is under development in the context of the SELFNET project [5], who has designed an architecture to implement this SON over NFV/SDN view, by integrating novel SDN, NFV, SON, Cloud computing artificial intelligence technologies [99]. In Fig. 5, we have represented a SON over SDN/NFV architecture, inspired by the work carried out by the SELFNET project [5]. The self-organizing capabilities are provided by two layers, the *SON control layer* and the *SON autonomic layer*. The SON control layer contains the applications that will enable the collection of data from sensors deployed throughout the network, as well as the applications that will enforce decisions towards elements of the network. The SON autonomic layer offers the mechanisms to provide network intelligence. It performs monitoring, collects information about the network behaviour, diagnose problems and makes decisions about what functions to activate, where to locate them, if at the edge or not, etc. As a result, automation is implemented at two levels. First inside the specific instantiated VNF, and then in the definition of the network itself, about where and how to locate the specifics VNFs.

In the market there already exist start-ups which advertise the concept of virtual-SON in the cloud. Cellwize [13] is one of them. They are promising a technology with deployment in the cloud, capable of working seamlessly across different vendors, spectrum and technologies.

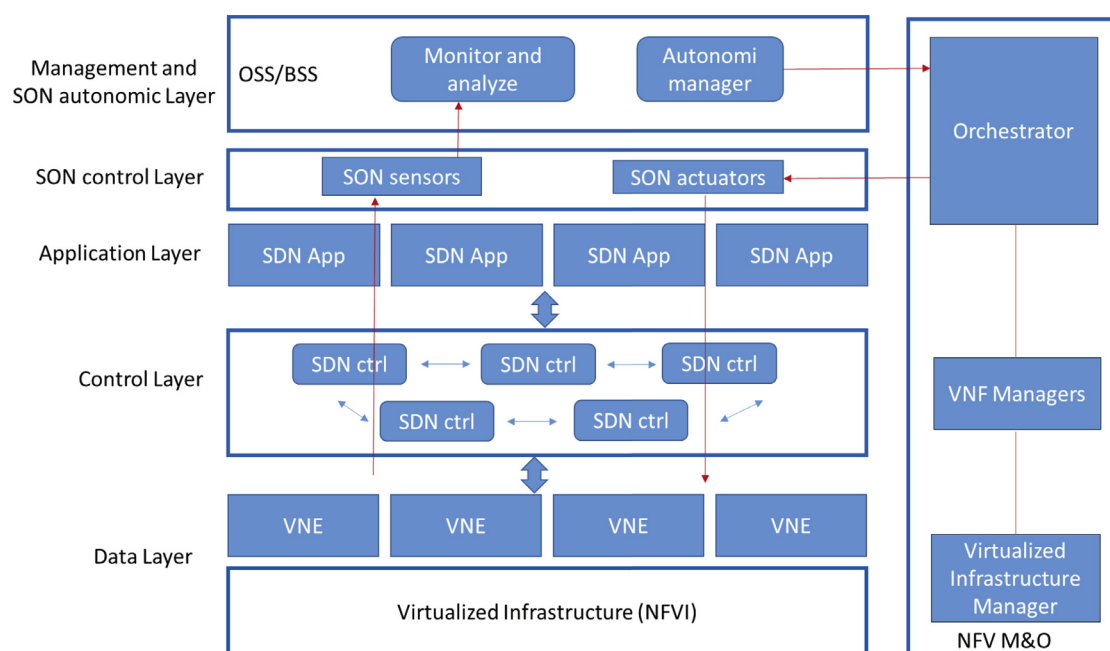


Fig. 5. SON over NFV-SDN architecture.



### 3. How to address SON and NM through ML

In this section we classify at high level the different network management classes of problems that one may need to deal with when aiming at managing the network in a self-organized manner. For each class of problem, we identify the machine learning tools that can be used. The objective of ML is to improve performance of a particular sets of tasks by creating a model that helps find patterns through learning algorithms. ML taxonomy is traditionally organized onto: 1. Supervised Learning (SL), 2. Unsupervised Learning (UL), 3. Reinforcement Learning (RL), 4. New trends: Deep Learning (see Table 2). Recently, new trends in the area of ML are taking momentum, thanks to the progress of software engineering, computational capabilities and memory availability. Deep learning has been proven feasible and extremely effective in different applications, like language, video, speech recognition, object and audio detection, among others. The most exemplary one is the win of AlphaGo, beating the world champion at the Chinese board game Go. The victory of AlphaGo was due to the implementation of a deep reinforcement learning algorithm capable of self-learning.

Keeping in mind the SON and NM functions introduced in the previous section, the classes of problems that need to be addressed when managing the network autonomously are:

- *Variable estimation or classification*: The tasks belonging to this class of problem aim at e.g. estimating the QoS or the QoE of the network, at predicting performances or behaviours of the network, by learning from the analysis of data obtained from past behaviours of the network. NM and SON functions where these tasks are useful are QoS estimation and other MDT use cases, the prediction of behaviours to optimize network parameters, etc. Solutions to these problems can be translated into finding the relationship between one variable and some others, or Identifying which class of a set of pre-defined classes the data belongs to. Solutions are then to be found in the SL literature, with both regression and classification tasks.
- *Diagnosis of network faults or misbehaviours*: The tasks belonging to this class of problems aim at detecting issues ongoing in the network, which may be associated to faults and anomalous setting of network parameters. This kind of problems relates to self-healing issues and solutions can be found in UL literature, and in particular in the anomaly detection solution.
- *Dimensionality reduction*: The network generates continuously a huge

amount of data. For an appropriate processing and to extract useful information, it is convenient to eliminate the noise present in the data base, by reducing the dimensionality of data. Solutions to this problem are to be found in the UL literature, and specifically among the dimensionality reduction solutions.

- *Pattern identification, grouping*: The tasks belonging to this class aim at identifying patterns, group of nodes with similar characteristics, according to some kind of criteria. An objective may be to apply to them similar optimization approaches. Self-configuration use cases are intuitive application for these issues. Solutions to these problems can be translated into learning the set of classes the data belongs to. UL literature offers solutions in the area of clustering.
- *Sequential decision problems for online parameter adjustment*: This class of problems is extremely common in the area of autonomous management, where we face control decision problems to online adjust network parameters, with the objective to meet certain performance targets. This kind of decision problems, where we learn the most appropriate decision online, based on the reaction of the environment to the actions the network is taking, can be addressed through RL solutions. All self-optimization use cases can be addressed through these solutions, as well as COC problems.

In the rest of the section, we relate each class of NM problem to the possible ML literature to solve it. The review of ML literature provided in the following, is far from being exhaustive. Many methods and techniques will not be described, because the purpose is here to provide a useful taxonomy to address NM and SON problems and to analyze and understand the related literature using ML solutions. For a deeper understanding of ML solutions, the reader is referred to more specific literature.

#### 3.1. Supervised learning (SL)

This ML technique could be extremely useful when the NM function to address requires estimation, prediction, classification of variables. SL is a ML technique which takes training data (organized into an input vector ( $\mathbf{x}$ ) and a desired output value ( $y$ )) to develop a predictive model, by inferring a function  $f(\mathbf{x})$ , returning the predicted output  $\hat{y}$ . For that, the construction of a dataset is needed. The dataset contains training samples (rows), and features (columns), and is usually divided into 2 sets. The training set, used to train the model, and the test set, used to make sure that the predictions are correct. The goal of the training model is to minimize the error between the predictions and the actual values. Hence, by applying ML, we aim to estimate how well a learning algorithm generalizes beyond the samples in the training set. The input space is represented by a  $n$ -dimensional input vector  $\mathbf{x} = (x^{(1)}, \dots, x^{(n)})^T \in \mathbf{R}^n$ . Each dimension is an input variable. In addition a training set involves  $m$  training samples  $((\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m))$ . Each sample consists of an input vector  $\mathbf{x}_i$ , and a corresponding output  $y_i$ . Hence  $x_i^{(j)}$  is the value of the input variable  $x^{(j)}$  in training sample  $i$ , and the error is usually computed via  $|\hat{y}_i - y_i|$ . The SL technique has two main applications, classification and regression. On the one hand, classification is applied when  $y$ , the output value we try to predict is discrete, e.g., we want to predict if a cancer is benign or malign, based on a dataset constructed based on medical records, and collecting many features, e.g. tumour size, age, uniformity of cell size, uniformity of cell shape. On the other hand, a regression problem is applied when  $y$  is a real number.

A huge amount of SL algorithms for classification can be found in the literature, and a study to evaluate the performance of some of them can be found in [100]. In the following we briefly introduce the most common algorithms.

1. k-Nearest Neighbors (k-NN) can be used for classification and regression. k-NN is a non-linear method where the input consists of the

**Table 2**  
Summary of selected ML techniques.

Supervised learning	
Classification	K-Nearest Neighbours
	Generalized Linear Model
	Support Vector Machines
	Naive Bayes
	Neural Networks
Regression	K-Nearest Neighbours
	Generalized Linear Regression
	Support Vector Regression
	Neural Networks
	Decision Trees
Unsupervised learning	
Clustering	Non-overlapping clustering
	Hierarchical clustering
	Overlapping clustering
Dimensionality Reduction	Feature Extraction
	Feature Selection
Anomaly Detection	Pruning techniques
	Rule-based systems
Reinforcement learning	
Model-based	Dynamic Programming
	Monte Carlo
Model-free	Temporal Difference

$k$  closest training samples in the input space. The predicted output is the average of the values of its  $k$  nearest neighbours. A commonly used distance metric for continuous variables is the Euclidean distance. The  $k$ -NN method has the advantage of being easy to interpret, fast in training, and the amount of parameter tuning is minimal. However, the accuracy of the prediction is generally limited.

2. Generalized Linear Models (GLM). The linear model describes a linear relationship between the output and one or more input variables, and where the approximation function maps from  $x_i$  to  $y_i$  as follows,

$$\hat{y}_i = \theta_0 + \theta_1 x_i^{(1)} + \dots + \theta_n x_i^{(n)} \quad (1)$$

where  $\theta_i$  are the unknown parameters. The idea is to choose  $\theta_i$  so that  $\hat{y}_i$  minimizes the loss function. Typically, we make the assumption that the samples in each dataset are independent from each other, and that the training set and testing set are identically distributed. Note that if the relation is not linear, the model should be generalized, in an attempt to capture this relationship [101].

3. Naive Bayesian. The method is used for classification and is based on Bayes theorem, i.e., calculating probabilities based on the prior probability. The main task is to classify new data points as they arrive. A NB classifier assumes that all attributes are conditionally independent, and is recommended when the dimensionality of the input is high [102]. Since NB assumes independent variables, it only requires a small amount of training data to estimate the means and variances of the variables.
4. Support Vector Machines (SVMs) can be used for classification and regression. SVMs are inspired by statistical learning theory, which is a powerful tool for estimating multidimensional functions [103,104]. This method can be formulated as a mathematical optimization problem, which can be solved by known techniques. For this problem, given  $m$  training samples  $((x_1, y_1), \dots, (x_m, y_m))$ , the goal is to learn the parameters of a function which best fit the data. It samples hyperplanes. Thus, the hyperplane with the main minimum distance from the sample points is maintained. The sample points that form margin are called support vectors and establish the final model. This method in general shows high accuracy in the prediction, and it can also behave very well with non-linear problems when using appropriate kernel methods. Also, when we cannot find a good linear separator, kernel techniques are used to project data points into a higher dimensional space where they can become linearly separable. Hence the correct choice of kernel parameters is crucial for obtaining good results. In practice, this means that an exhaustive search must be conducted on the parameter space, thus complicating the task [105].
5. Artificial Neural Network (ANN) is a statistical learning model inspired by the structure of a human brain, where the interconnected nodes represent the neurons producing appropriate responses. ANN supports both classification and regression algorithms. The basic idea is to efficiently train and validate a neural network. Then, the trained network is used to make a prediction on the test set. In this method the weights are the parameters in charge of manipulating the data in the calculations. Here, the interconnection pattern between the different layers of neurons, the learning process for updating the weights of the interconnections, and the activation function that converts a neuron's weighted input to its output activation are the most important parameters to be trained [106]. ANNs methods require parameters or distribution models derived from the data set, and in general they are also susceptible to over-fitting.
6. Decision Trees (DT) is a flow-chart model in which each internal node represents a test on an attribute. Each leaf node represents a response, and the branch represents the outcome of the test [107]. DTs can be used for classification and regression, and they have nuisance parameters, such as the desired depth and number of

leaves in the tree [108]. Also, they do not require any prior knowledge of the data, are robust (i.e., do not suffer the curse of dimensionality as they focus on the salient attributes) and work well on noisy data. However, DTs are dependent on the coverage of the training data as with many classifiers. Moreover, they are also susceptible to over-fitting.

Methodologies have also been proposed to take the best out of the available data, to boost the prediction performance. Some of these methodologies are classified among the so called *Ensemble methods*. Ensemble methods combine the predictions of multiple learning algorithms to produce a final prediction. This technique has been investigated in a huge variety of works [109,110]. A general method is sub-sampling the training examples, where the most useful techniques are referred to as bagging and boosting [111]. Bagging manipulates the training examples to generate multiple hypothesis. It runs the learning algorithm several times, each one with different subset of training samples. On the other hand, AdaBoost maintains a set of weights over the original training set, and adjusts these weights by increasing the weight of samples that are misclassified, and decrease the weight of samples that are correctly classified [112,113].

### 3.2. Unsupervised learning (UL)

This kind of learning can be extremely useful when the NM function requires identifying anomalous behaviours, recognizing patterns or reducing the dimensionality of the data. UL is a ML technique, which receives unlabelled input patterns with the objective to find a pattern in it. In this case, we let the computer learn by itself, without providing the correct answer to the problem we want to solve. The goal is to construct representation of inputs that can be used for predicting future inputs without giving the algorithm the right answer, as in turn we do in case of supervised learning [114]. The three most important families of algorithms are clustering, dimensionality reduction and anomaly detection techniques. There are many examples of UL applications in our daily life, e.g., news.google.com, understanding genomics, organize computer clusters, social network analysis, astronomical data analysis, market segmentation, etc. In the context of SON, UL algorithms are applied mainly on self-optimization and self-healing use cases.

1. Clustering. This technique aims at identifying groups of data to build representation of the input. The most common methods to create clusters by grouping the data are: non-overlapping, hierarchical and overlapping clustering methods. K-means [115] and Self-organizing Maps (SOMs) [116] methods belong to non-overlapping clustering techniques. When the clusters at one level are joined as clusters at the next level (cluster-tree), this is referred in literature as a hierarchical clustering method [117]. In case that an observation can exist in more than one cluster simultaneously, this is known as overlapping or fuzzy clustering. Fuzzy C-means and Gaussian mixture models belong to this kind of technique [115,118]. Also Hidden Markov Model (HMM) can be used for clustering. This kind of algorithms have been proposed in a wide range of fields, such as, robotics, wireless systems, and routing algorithms for mobile ad-hoc networks, among others.
2. Dimensionality Reduction. High-dimensional datasets present many challenges. One of the problems is that, in many cases, not all the measured variables are necessary to understand the problem of interest. In the state of the art we can find a huge amount of algorithms to predict models with good performance from high-dimensional data. However it is of interest for many problems to reduce the dimension of the original data. For example, in [119,120], the authors face the problem of the huge amount of potential features the system has as input, and they suggest that the regression analysis has a better performance in a reduced space. In this context, the most common methods are: Feature Extraction (FE) and Feature

Selection (FS) [121]. Both methods seek to reduce the number of features in the dataset. FE methods do so by creating new combinations of features (e.g. Principal component analysis (PCA)), which project the data onto a lower dimensional subspace by identifying correlated features in the data distribution. They retain the Principal Components (PCs) with the greatest variance and discard all others to preserve maximum information and retain minimal redundancy [122]. Correlation based FS methods include and exclude features present in the data without changing them. For example, Sparse Principal Component Analysis (SPCA) extends the classic method of PCA for the reduction of dimensionality of data by adding sparsity constraint on the input features.

3. Anomaly Detection. Anomaly detection identifies events that do not correspond to an expected pattern. By modeling the most common behaviors, the machine selects the set of unusual events [123]. Self healing is one of the main functionality in which this kind of techniques are applied, some examples are [124,125]. The two most common techniques are: Rule based systems, which are very similar to DTs, but they are more flexible than DTs as new rules may be added, without creating a conflict with the existing ones [123] and Pruning techniques, which aim at identifying outliers, where there are errors in any combination of variables.

### 3.3. Reinforcement learning (RL)

The ML approaches under this category can be used to address NM functions which require network parameter control. Differently from the case of SL, RL aims to learn from interactions how to achieve a certain goal. In many real applications and in particular, in sequential decision and control problems, it is not possible to provide an explicit supervision to the training (i.e. the right answer to the problem). In these cases, we can only provide a reward/cost function, which indicates to the algorithm when it is doing well and when it is doing poorly. RL has already been proven effective in many real world applications, such as autonomous helicopters, network routing, robot legged automation, etc. [126–128].

The learner or decision maker is called *agent*, and it interacts continuously with the so-called *environment*. The agent selects actions and the environment responds to those actions and evolves into new situations. In particular, the environment responds to the actions through *rewards*, i.e., numerical values that the agent tries to maximize over time.

The agent has to exploit what it already knows in order to obtain a positive reward, but it also has to explore in order to take better actions in the future. Learning can be centralized in a single agent or distributed across a multiple agents. In single agent systems, ML approaches are capable of finding optimal decision policies in dynamic scenarios with only one decision maker. In multi-agent systems, the distributed decisions are made by multiple intelligent decision makers, and the optimal solutions or equilibria are not always guaranteed [129].

The problem is then defined by means of a Markov Decision Process (MDP)  $\{S, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma\}$ , where  $S$ , is the set of possible states of the environment  $S = \{s_1, s_2, \dots, s_n\}$ ,  $\mathcal{A}$ , is the set of possible actions  $\mathcal{A} = \{a_1, a_2, \dots, a_q\}$  that each decision maker may choose,  $\mathcal{T}(s'|s, a)$ , is the transition function denoting the probability of getting  $s'$  giving an action  $a$  in state  $s$ ,  $\mathcal{R}(s, a)$  is a reward function, which specifies the expected immediate return obtained by executing action  $a$  in state  $s$ , and  $0 \leq \gamma \leq 1$  is a discount factor, which gives more importance to immediate rewards compared to rewards obtained in the future [130].

The MDP represents the theoretical basis for the RL framework [130]. At each time step, the agent implements a mapping from states to probabilities of selecting each possible action. This mapping is the agent's *policy*.

The objective of each learning process is to find an optimal policy  $\pi^*(s) \in \mathcal{A}$  for each  $s$ , to maximize some cumulative measure of the reward  $r$  received over time. Almost all RL algorithms are based on

estimating a so called *value function*, which is a function of the states estimating how good it is for an agent to be in a given state. For MDPs the state-value function, denoted as  $V_\pi(s)$ , is the expected return when starting in state  $s$  and following policy  $\pi$  thereafter. For more information the reader is referred to [130].

RL literature offers two approaches to solve MDPs. These two approaches are: model-based and model-free.

1. Model-based. Dynamic Programming (DP) and Monte Carlo (MC) methods fall into the category of model-based approach.

- (a) DP is able to solve MDPs by relying on the knowledge of the state transition probability between two states after executing a certain action. DP is an algorithmic paradigm that solves a given complex problem by breaking it into sub-problems and stores the results of sub-problems to avoid computing the same results again. DP algorithms are based on update rules derived from the Bellman equation. The first key component is known as the *policy evaluation* process, according to which a policy provides information about how much reward is going to be received in the MDP. This solution is used to build the first overall solution by finding the optimal policy known as the *policy iteration* process. Finally the *value iteration* makes the value function better and better by applying Bellman's equation intuitively. DP is used to solve problems such as, scheduling, graph algorithms, bioinformatics, among others.

- (b) MC method only requires experience, i.e., sample sequences of states, actions and rewards. The estimations are only updated after the episodes conclude. Although their application on practical cases is limited, they provide foundation for other RL methods.

2. Model-free. Temporal Difference (TD) methods are model free approaches to solve RL problems. TD learning is a combination of MC and DP ideas. It uses the current estimate  $V_t^\pi$  of the value function instead of the exact  $V^\pi$ , as it happens in DP. If  $\mathcal{T}$  is known, we can solve the MDP through DP, otherwise we need to rely on TD methods.

Some common examples of TD methods are: Q-learning, Sarsa and Actor Critic (AC) [130]. TD methods can be found in each SON functionality.

- (a) Q-learning and Sarsa are based on the estimation of the state-action value function,  $Q(s, a)$ . Learning is performed by iteratively updating the Q-values, which represent the expert knowledge of the agent, and have to be stored in a representation mechanism. The most intuitive and common representation mechanism is the lookup table, i.e., the TD methods represent their Q-values in a Q-table, whose dimension depends on the size of the state and action sets. The difference between them is that, Q-learning is an off-policy learner. This means that, the agent will use the policy corresponding to the best action in the next state, given the current agent experience, whereas Sarsa is an on-policy learner. On-policy learners evaluate the policy  $\pi$ , to perform the decisions. This means that, the policy followed by the agent to select its behaviour in a given state is the same used to select the action based on which it evaluates the followed behaviour.

- (b) AC methods have a separate memory structure to represent the policy independently of the value function. The policy structure is known as the *actor*, since it is used to select the actions, while the estimated value function is known as the *critic*. The critic learns and critiques whatever policy is currently being followed by the actor and takes the form of a TD error  $\delta$ , which is used to determine if  $a_t$  was a good action or not.  $\delta$  is a scalar signal, which is the output of the critic and drives the learning procedure. After each action selection, the critic evaluates the new state to determine whether things have gone better or worse than expected.

### 3.4. New trends: deep learning

Deep learning is a new trend in ML that allows computer systems to improve with experience and data. It achieves great power and flexibility to operate in complicated real-world environments, by learning to represent the world through a nested hierarchy of concepts. The ML algorithms that we have reviewed in this paper have a strong dependency on the features on which the algorithms are applied. Based on that, much effort has been devoted to design ML algorithms that yield to useful representations. This is known as *representation learning*, and deep learning is one way of learning representations [131,132]. The main representation of deep learning is through a Multilayer Perceptron, which is a multilayer neural network function mapping some sets of input values to output values. Each layer of this representation learns a hierarchy of the output. Deep Learning has the ability to do successful training from the bottom layers to the higher ones. This is done by applying computational models that are composed of multiple levels of representation and abstraction that help make sense of data.

## 4. Machine learning enabled network management

As we have mentioned in the introduction of this work, mobile networks constitute a huge source of data which could be analyzed with proper tools, with the primary goal to make more informed decisions when it comes to efficiently manage the overall 4G or 5G network. In this context, ML is a great opportunity due to its capability of providing insightful information from the analysis of data already available to operators, which can be used to make improvements or changes.

In this section we focus on how ML can specifically be applied to SON and novel network management concepts. First, we present all the relevant sources of information that could be extracted from a mobile network. All these data are available to operators, and may happen to be sensitive data for the users' privacy. However, some interesting data can be derived from open databases or sniffed from unencrypted control channels like the PDCCH. We will then discuss on these options. Second, we will go through again the main SON and network management functions and provide a classification of the main inputs and outputs that we would need available in the form of data, when designing an appropriate ML algorithm to target the specific use case, and the KPI indicators that we would need to monitor. Finally, we provide an overview of SON and network management's related work, where ML techniques have been adopted, classifying this work as a function of the targeted use case, the specific high level problem to solve and the ML technique that the authors have picked to address the problem.

### 4.1. Data generated by mobile cellular networks

As we observed in [23] a huge amount of data is currently already generated in mobile networks during normal operations by control and management functions. This kind of data can be exploited to find patterns and extract useful information from them. This allows to take more informed decisions to effectively manage network performance. Some examples of the different sources information generated by mobile networks, together with the kind of usage currently provided by operators, and related references of interest, is detailed in Table 3.

1. Charging Data Records (CDR). They are defined in [133] and provide a comprehensive set of statistics at the service, bearer and IMS levels. These records are typically stored for offline processing by the operator. The granularity of this information in the time domain is however quite coarse, as records are generated in correspondence with high-level service events (e.g., start of a call).
2. Performance management functionality. This data source [42,134] provides data regarding the network performance and it covers, among others, aspects of the performance of the radio access network, such as, radio resource control and utilization, performance of

the various bearers (both on the radio part and in the back-haul), idle and connected mode mobility.

3. Minimization of Drive Tests (MDT). The data extracted from this source refers to the radio measurements of both idle and connected mode mobility, coverage items, such as, power measurements and radio link failure events, and can be associated with position information of the UE performing the measurement. More information on these data has already been provided also in Section 2.7.
4. E-UTRA Control plane protocols and interfaces, such as Radio Resource Control (RRC), S1-AP, X2-AP protocols, are another huge source of information, especially concerning aspects, such as cell coverage, user connectivity, mobility in idle and connected mode, inter-cell interference, resource management, load balancing, among others.
5. Data plane traffic flow statistics, also are a huge source of information, which can be gathered at various points of the network, like the eNB, or the PDN Gateway (PGW) and Serving Gateway (SGW). The Internet Protocol Flow Information Export (IPFIX) is an example of standardized format to exchange this kind of statistics [136].

All these data are available to the network operators, but in most cases this is not made available to the academic community due to privacy issues and network operators' interests. There are some exceptions, like the *Data for Development* (D4D) initiative from the Orange group [137], which made available anonymous data extracted from the Senegal's network to research laboratories. However these data are in general aggregated and do not allow deep insight into the operator's network.

This lack of data represents a great limitation for the advancement of the ML based network management research. It is not easy to find contributions analyzing real network management data. Some work can be found in the context of 3G networks, but for 4G networks it is very hard to find works considering real data [138,139]. These works though do not take into account the data analyzing them through ML techniques, to extract experience from them.

However, some network data can be derived in other ways. Some databases are available, providing a huge insight in mobile network operators. Some examples are listed in the following, together with information that can be extracted from them.

- **opencellid**: It contains information on specific cells, such as: network type (GSM, UMTS, LTE), Mobile Country Code (MCC), Mobile Network Code (MNC), Location Area Code (LAC) for GSM and UMTS, Tracking Area Code (TAC) for LTE, Cell ID (CID) for GSM and LTE networks, Primary Scrambling Code (PSC) for UMTS networks, PCI for LTE networks, longitude and latitude in degrees, estimates of range in meters, total number of measurements collected from the tower, defines if the coordinates of the cell tower are exact or estimated, information of the date when the cell tower was first added to the data base and updated, average signal strength from all measurements received from the cell in dBm, or as defined in [140]. This data base also receives funding from important vendors like Qualcomm [141] and offers some formula of free access to portion of data for academic purposes.
- **opensignal**: It offers information on achievable data rates, latency and availability, per operators, but not information per cell tower [142].
- **antenasgsm**: It offers information on maps and positions of cells, with added information on the operator and the assigned bandwidth [143].
- **Google geolocations API**: It allows queries based on the cell ID to get cell related information and WiFi Access Points (APs), such as latitude and longitude [144].

The information provided by these databases is precious, but still does not give sufficient insight on the behaviour of the network, and mainly offers an overview of the coverage provided by the single operators.



**Table 3**  
Information elements relevant for ML enabled SONs.

Source	Data	Usage	TS
Charging Data Records (CDR)	Includes statistics at the service, bearer and IP Multimedia System (IMS) levels.	These records are typically stored, but only used by customer service. The network operation departments typically do not leverage this information and do not have access to it, as much as customer service does not leverage network management data.	TS 32.298 [133]
Performance management (data on network performance)	It covers long-term network operation functionalities, such as Fault, Configuration, Accounting, Performance and Security management (FCAPS), as well as customer and terminal management. An example is that defined for OAM, which consists of aggregated statistics on network performance, such as number of active users, active bearers, successful/failed handover events, etc. per BS, as well as information gathered by means of active probing.	The data is currently mostly used for fault identification, e.g., triggering alarms when some performance indicator passes some threshold, so that an engineer can investigate and fix the problem. Typically, the only automatic use of this info is threshold-based triggering, which can be done with very low computational complexity.	TS32.401 [134], TS32.425 [42]
Minimization of Drive Tests (MDT)	Radio measurements for coverage, capacity, mobility optimization, QoS optimization/verification	This data is used for identified use cases such as coverage, mobility and capacity optimization, and QoS verification	TS37.320 [135]
E-UTRA Control plane protocols and interfaces	Control information related to regular short-term network operation, covering functionalities such as call/session set-up, release and maintenance, security, QoS, idle and connected mode mobility, and radio resource control.	A This information is normally discarded after network operation purposes have been fulfilled. Some data can be gathered via tracing functionality or used by SON algorithms which normally discards the information after usage	TS36.331 [54], TS36.413 [45], TS36.423 [46]

To get more information, still we can do something more and access directly to the unencrypted PDCCH and extract information exchanged between the users and the associated eNB. In particular, it is possible to build a sniffer, as the one described in [145], from which to collect raw communication traces exchanged by the users and the associated eNodeB. This allows to have access not only to aggregate base station statistics, but also to more valuable information derived from the radio protocols, such as the resource block allocation and the link adaptation mechanism of the system. In particular, the OWL sniffer [145] is an online decoder of the LTE control channel, which uses a Software Defined Radio (SDR) to send the raw LTE signal to a PC running the decoding software. This open-source software is capable of reliably logging the LTE downlink control information (DCI) broadcasted by base stations. In fact, LTE uses an unencrypted control channel to assign network resources to users for both downlink and uplink communications. Resources are assigned to devices through their radio network temporary identifiers (RNTIs), every millisecond, specifying the number of resource blocks (RBs) and the modulation and coding scheme (MCS) to be used. There are works in literature using this sniffer to collect and analyze traces from different European cities [146].

Another option is to use a high-fidelity network simulator like ns-3 LTE/LTEPC Network Simulator (LENA) module, to generate realistic data [147]. This simulator has been built around industrial Application Programming Interface (API) defined by the small cell forum and offers high-fidelity models from Media Access Control (MAC) to application layers. It has also been designed with the requirement to simulate tens of eNBs and hundreds of UEs, and to specifically test Radio Resource

Management (RRM) and SON algorithms. Consequently it could be a very useful tool to build realistic scenarios based on information available in public databases, generate data to analyze, build algorithms based on this analysis and close the loop on the simulator to test the designed algorithms. The lack of data to analyze when designing future networks will always be a challenge also for vendors, in the phase that the networks are still not running, as a result, relying on high fidelity open source simulators, like those provided by ns-3 consortium, available to anyone and allowing collaborative development, and results reproducibility, is a fundamental aspect of research in this line.

Finally, let us review the main SON use cases in Table 4, by analyzing the main input information that their design would require, in terms of data, together with the main identified output actions and meaningful associated KPIs.

#### 4.2. Overview of ML based network management's relevant literature

This section reviews SON and Network management's recent work in the area of ML. We will go through each main function and use case and review significant literature and ML approach that has been used to approach the problem. Table 5 summarizes the main works in this area and classifies them per 3GPP use case, technique and specific algorithm adopted by the authors.

1. Use case: Indicates the 3GPP targeted use case.
2. Reference: Indicates the reference of the related work.
3. Technique: Indicates the applied ML method (Supervised Learning,

**Table 4**  
SON inputs, outputs and KPIs.

SON function	Inputs	Output actions	KPIs
MLB	X2 resource status and load estimation information.	Tuning the CIO, i.e. offsets of serving and neighbour cells to satisfy handover conditions.	Improved QoS and capacity
MRO	S1AP and X2AP handover requests, handover reports, RLF reports and indications.	A3 offsets, TTT, L1 and L3 filter coefficients, in connected mode, and Qoffset in Idle mode.	Minimized call drops, RLFs and ping pong effects.
CCO	UE measurements	Transmission power, pilot power, antenna parameters, coordinated Almost Blank Subframes (ABS)	Maximized coverage and cell and edge throughput
ICIC	HII, RNTP, OI, UE Measurements.	Transmission power, pilot power, antenna parameters, coordinated ABS	Minimized Inter-cell interference.
COC	UE Measurements.	Transmission power, antenna parameters of neighbouring cells	Minimized outage.
ES	Resource status, UE Measurements.	Switch ON and OFF policies	Minimized energy consumption.

- Unsupervised Learning, Reinforcement Learning).
4. Problem: Indicates the general problem to solve.
  5. Algorithms: Indicates the specific ML algorithm applied to the data (see Table 5).

#### 4.2.1. Mobility load balancing

The literature offers some examples of application of ML techniques to the MLB use case. The majority of applications fall in the area of RL, as the main problem to solve is a sequential decision problem about how to set configuration parameters, which optimize network performance and user experience. An example of a RL application for MLB use case can be found in [149]. Here the authors present a distributed Q-learning approach that learns for each load state the best MLB action to take, while also minimizing the degradation in HO metrics. Another option to take advantage also of fuzzy logic capabilities of dealing with heterogeneous sources of information is provided in [150], where fuzzy logic is combined with Q-learning in order to target the load balancing problem. For similar reasons, fuzzy logic is also proposed in [151] to enhance the network performance by tuning HO parameters at the

adjacent cells. Approaches incorporating fuzzy logic with RL capabilities have the advantage to capture the uncertainty existing in real world complex scenarios, while schemes considering only learning approaches may be limited by the fixed variable definition. When combining fuzzy logic with RL, also the subjectivity with which the fuzzy variable may be defined is overcome by the adjusting capabilities of the learning. Alternatively, a centralized solution is approached in [152], where a central server in the cellular network determines all HO margins among cells by means of a dynamic programming approach. Besides RL, also clustering schemes have been proposed in this area, to group cells with similar characteristics and provide for them similar configuration parameters [153]. Considering clustering in large realistic scenarios is an added value to reduce computational complexity and take advantage of what is learnt in other regions of the network where we observe similar environment characteristics.

#### 4.2.2. Mobility robustness optimization

Also for the case of MRO, we find in literature different solutions based on RL to solve a control decision problem. In [155,156], the

**Table 5**  
Related work.

	Reference	ML technique	Problem	Algorithm
<b>Self-configuration</b>				
PCI	[148]	UL	Planning	Clustering
<b>Self-optimization</b>				
MLB	[149]	RL	Control optimization	Q-learning
	[150]	RL	Control optimization	Q-learning
	[151]	RL	Control optimization	Fuzzy Q-learning
	[152]	RL	Control optimization	Dynamic Programming
	[153]	UL	Grouping	K-means clustering
	[154]	SL	Prediction	Multivariate polynomial regression
MRO	[155]	RL	Control optimization	Q-learning
	[156]	RL	Control optimization	Q-learning
	[157]	RL	Control optimization	Fuzzy control
	[158,159]	UL	Pattern identification	SOM
	[160]	UL	Prediction	Semi-Markov model
	[161–164]	SL	Prediction	ANN
CCO	[165]	RL	Control optimization	Fuzzy Q-learning
	[166]	RL	Control optimization	Fuzzy Q-learning
	[167]	RL	Control optimization	Fuzzy Q-Learning
	[168]	UL, RL	Control optimization	Fuzzy ANN/Q-learning
ICIC	[169]	RL	Control optimization	Q-learning
	[170]	RL	Control optimization	Fuzzy Q-learning
	[171,172]	RL	Control optimization	Q-learning
ES	[173]	RL	Control optimization	Q-learning
	[174]	UL	Decision making	Fuzzy logic
	[175,176]	UL	Grouping, pattern identification	Clustering
<b>Self-healing</b>				
COC	[177]	RL	Control optimization	Actor Critic
	[178]	RL	Control optimization	Actor-Critic
	[179]	SL	Control optimization	Fuzzy logic
COD	[180]	UL	Anomaly detection	Diffusion Maps
	[181]	SL	Anomaly detection	Fuzzy logic
	[182]	SL/UL	Diagnosis	Naive Bayesian
	[183]	SL	Anomaly detection	SVM, Ensemble methods
	[178]	SL/UL	Anomaly detection	k-NN, local-outlier-factor
	[184]	UL	Grouping, pattern identification	Hidden Markov Model
	[185–187]	SL	Fault Detection	k-NN
	[188,189]	SL	Diagnosis	Naive Bayesian
<b>Self-coordination</b>				
	[190]	SL	Classification	Decision Trees
	[191]	RL	Control optimization	Actor Critic
	[192]	RL	Control optimization	Q-learning
	[193]	RL	Control optimization	Actor Critic
	[194]	SL/UL	Prediction	Bagged-SVM
<b>MDT</b>				
	[195,196]	SL	Verification/estimation	Linear correlation
	[119]	SL	Prediction	Regression models
	[197]	SL/UL	Prediction/curse of dimensionality	Regression models/DR
	[120,198]	SL	Prediction	Bagged-SVM/DR
<b>Core Networks</b>				
	[199]	SL	Prediction	Adaboost, SVM

authors focus on the optimization of the users' experience and of the HO performance. In [155] the authors take advantage of the Q-learning approach to effectively reduce the call drop rates, whereas in [156], unlike other solutions that assume a general constant mobility, the authors adjust the HO settings in response to the mobility changes in the network by means of a distributive cooperative Q-learning. Differently from [155,156], in [157], the authors take advantage also of fuzzy logic capabilities. These solutions are based on control optimization of HO parameters through RL, so they propose similar solutions to those found in the literature of MLB. In this case we can do the same considerations about the advantages of considering fuzzy logic in order to gain in flexibility in the uncertain and complex real network context. Different approaches in turn, address the problem by identifying successful HO events, through solutions based on unsupervised learning. In particular, the works of [158] and [159] propose an approach to HO management based on UL and SOM analysis. The idea is to exploit the experience gained from the analysis of data of the network based on the angle of arrival and the received signal strength of the user, to learn specific locations where HOs have occurred and decide whether to allow or forbid certain handovers to enhance the network performance. The solutions enable self-tuning of HO parameters to learn optimal parameters' adaptation policies. Similarly, in [160] the authors exploit the huge amount of information generated in the network to predict user traffic distribution. In particular, they take advantage of semi-Markov model for spatiotemporal mobility prediction in cellular networks. Finally, the works in [161–164], propose schemes to make predictions about UE's mobility, which allows to anticipate smart HO decisions.

#### 4.2.3. Coverage and capacity optimization

In case of CCO, different approaches in literature focus on RL solutions based on continuous interactions with the environment, oriented to online adjusting antenna tilts and transmission power levels through TD learning approaches. In [165] and [166] a fuzzy Q-learning approach to optimize the complex wireless network, by learning the optimal antenna tilt control policy has been proposed, and a similar approach is followed also in [167] and [168]. In addition, they also propose to combine fuzzy logic with Q-learning, in order to deal with continuous input and output variables [167]. also proposes a central control mechanism, which is responsible to initiate and terminate the learning optimization process of every learning agent deployed in each eNB. Finally, [168] innovates with respect to other approaches since in order to adjust the antenna tilt and transmission power parameters, it considers the load distribution of the different cells involved in the optimization process, and introduces novel mechanisms to facilitate cooperative learning among the different SON entities.

#### 4.2.4. Inter-cell interference coordination

Similarly to the CCO case, ML has been proposed in the literature of ICIC use case as a valid solution, where RL is the principle used tool, with special emphasis to TD methods, in order to target the optimization of control parameters. Several works target the problem to minimize the interference among cells by using the most common TD learning method, Q-learning [169–172]. The work in [169] is related to control inter-cell interference in a heterogeneous femto-macro network. The work combines information handled by the multi-user scheduling with decisions taken by a learning agent based on Q-learning, which tries to control the cross-tier interference per resource block. [170] proposes a distributed solution for ICIC in OFDMA networks based on a Fuzzy Q-learning implementation. The proposed solution achieves joint improvement for all users, i.e., the improvements of users with bad quality do not come at the expense of users with good quality. Moreover, a decentralized Q-learning framework for interference management in small cells is proposed in [171]. The authors focus on a use case in which the small cell networks aim to mitigate the interference caused to the macro-cell network, while maximizing their own spectral

efficiencies. Finally, in [172] also a decentralized Q-learning approach for interference management is presented. The goal is to improve the systems performance of a macro-cellular network overlaid by femto-cells. In order to improve the time of convergence, a mitigation approach has been introduced, allowing them to have significant gains in terms of throughput for both, macro and femto-users. Interesting trade-offs can be studied to compare centralized vs. distributed solutions. In the novel context of small cells distributed solutions to interference management are to be preferred over more complex centralized solutions, but convergence and instability approaches may appear to affect the TD learning schemes, compromising system performances [169].

#### 4.2.5. Energy savings

Energy saving schemes for wireless cellular systems have been proposed in the past, enabling cells to go into a sleep mode, in which they consume a reduced amount of energy. In order to reduce the energy consumption of the eNBs, we can find several works related to ML techniques. An example can be found in [173], where the authors take advantage of RL to propose a decentralized Q-learning approach to allow energy savings by learning a policy by the iterations with the environment taking into account different aspects over time, such as the daily solar irradiation. Also, in [174], the authors switch off some underutilized cells during off peak hours. The proposed approach optimizes the number of base stations in dense LTE pico-cell deployments in order to maximize the energy saving. For the purpose, they use a combination of Fuzzy Logic, Grey Relational Analysis and Analytic Hierarchy Process tools to trigger the switch off actions, and jointly consider multiple decision inputs for each cell. This last work uses smart decision theory approaches, which though are not able to take advantage of the previous decisions made in the same environment, as in turn does the work proposed in [173], as a result of the TD learning approach. This allows that the work in [173] offers a more solid solution, considering also past information in the decision. Also for HetNets, we find several works, such as, [175,176], where the authors take advantage of KPIs available in the network for the construction of different kind of databases to analyze the potential gains that can be achieved in clustered small cell deployments.

#### 4.2.6. Cell outage compensation

The literature already offers different works targeting the problem of COC. For this use case RL has been proven as a valid solution since it is a continuous decision making/control problem. In this context a contribution in the area of self-healing has been presented in [177,178], where the authors present a complete solution for the automatic mitigation of the degradation effect of the outage by appropriately adjusting suitable radio parameters of the surrounding cells. The solution consists of optimizing the coverage and capacity of the identified outage zone, by adjusting the gain of the antenna due to the electrical tilt and the downlink transmission power of the surrounding eNBs. To implement this approach, the authors propose a RL based on actor-critic theory to take advantage of its capability of making online decisions at each eNB, and of providing decisions adapting to the evolution of the scenario in terms of mobility of users, shadowing, etc., and of the decisions made by the surrounding nodes to solve the same problem. A COC contribution also based on ML is targeted in [179], where fuzzy logic is proposed as the driving techniques to fill a coverage gap. The authors show performance gains by using different parameters, such as, the power transmission, the antenna tilt, and a combination of the two schemes. These two works are compared in [177] and the work in [177] is proven superior thanks to the ability to learn from the past experience introduced by the RL actor-critic approach.

#### 4.2.7. Cell outage detection

As we already mentioned, COD aims to autonomously detect cells that are not operating properly due to possible failures. For this kind of problem, anomaly detection algorithms offer an interesting solution

that allows to identify outliers measurements, which can be highlighting a hidden problem in the network. Proposals of solutions for this problem can be found in [180] and [181]. In particular, [180] presents a solution based on diffusion maps, by means of clustering schemes, capable of detecting anomalous behaviours generated by a sleeping cell. [181] presents a solution based on fuzzy logic for the automatic diagnosis of a troubleshooting system. In order to determine if there is a failure, the authors propose a controller, which receives as an inputs a set of representative KPIs. A similar approach is presented by [182], where the authors present an automated diagnosis model for Universal Mobile Telecommunications System (UMTS) networks based on Naive Bayesian classifier, and where the model uses both network simulator and real UMTS network measurements. In the context of this kind of classifiers, the works in [188,189], also take advantage of NB for automated diagnosis based on different inputs network performances. The work in [183] addresses both the case of outage and the one where in turn the cell can provide a certain level of service, which though does not allow to fulfill the expected UEs requirements. The approach relies on ensemble methods to train KPIs extracted by human operators to make informed decisions. In [200], the authors consider large data sets to identify anomaly behaving base station. They proposed an algorithm consisting of preprocessing, detection and analysis phases. The results show that by using dimensionality reduction and anomaly detection techniques irregularly behaving base stations can be detected in a self-organized manner. In [178] data gathered through MDT reports is used for anomaly detection purposes. Furthermore, the works of [185–187] take advantage of  $k$ -NN algorithm to propose a self-healing solution, in particular to tackle the fault detection domain. Finally, in [184], the authors consider a HetNet and they take advantage of HMM to automatically capture the dynamic's of four different states and probabilistically estimate if there exist a possible failure.

#### 4.2.8. SON conflicts coordination

As the deployment of stand-alone SON functions is increasing, the number of conflicts and dependencies between them also increases. Hence, an entity has been proposed for the coordination of this kind of conflicts. In this context, current literature includes several works based on ML. In [190] the authors focus on the classification of potential SON conflicts and on discussing the valid tools and procedures to implement a solid self-coordination framework. Q-learning, as a RL method, has been proposed in [191] to take advantage of experience gained in past decisions, in order to reduce the uncertainty associated with the impact of the SON coordinator decisions when picking an action over another to resolve conflicts. In [192], the authors use Q-learning to deal with the conflict resolution between two SON instances. Decision trees have been proposed in [201] to properly adjust Remote Electrical Tilt (RET) and transmission power. Additionally, in [193] the authors provide a functional architecture that can be used to deal with the conflicts generated by the concurrent execution of multiple SON functions. They show that the proposed approach is general enough to model all the SON functions and their derived conflicts. First they introduce these SON functions in the context of the general SON architecture, together with high-level examples of how they may interfere. Second, they define the state and action spaces of the global MDP that models the self-optimization procedure of the overall RAN segment. They show that the global self-optimization problem can be decomposed onto as many Markov decision sub-process (subMDP) as SON functions. Finally, the authors in [194] propose a scheme that provides a solution to deal with the potential conflicts between two of the most important SON functions in the context of mobility, namely MLB and MRO, which require the updating of the same set of handover parameters. The proposed scheme shifts the complexity to the creation of a prediction model that uses historical data and that allows to anticipate the network performance. Therefore, fast performance evaluations are allowed when the optimization is running.

#### 4.2.9. Minimization of drive tests

The great majority of literature using the MDT functionality to target MDT use cases, takes advantage of supervised and unsupervised learning techniques to provide different solutions for the different use cases. An example of that can be observed in [195,196], where the authors address the QoS estimation by selecting different KPIs and correlating them with common nodes measurements, to establish whether a UE is satisfied with the received QoS. A similar objective is targeted in [119], however, differently from the previous works, here the authors focus on multi-layer heterogeneous networks, so in a more complex and realistic scenario than the traditional macro-cell one. In particular, they present an approach, based on regression models, which allows to predict QoS in heterogeneous networks for UEs, independently of the physical location of the UE. This work is extended in [197] by taking into account the most promising regression models, but also analyzing dimensional reduction techniques. By doing PCA/SPCA on the input features, and promoting solutions in which only a small number of input features capture most of the variance, the number of random variables under consideration is reduced. Based on previous results, in [120,198] the same authors define a methodology to build a tool for smart and efficient network planning, based on QoS prediction derived by proper data analysis of UE measurements in the network.

Moreover, the work in [202] presents a system based on a fuzzy logic controller to improve network performances by adjusting antenna tilts values in a LTE system. Differently from previous works, the authors consider the use of call traces to identify the level of coverage, overshooting and overlapping problems, which are the inputs to the algorithm. Also, in [203], the same authors take advantage of connection traces (signal strength, traffic, and resource utilization measurements) to improve the network infrastructure in terms of spectral efficiency. The proposed method is designed to be integrated in commercial network planning tool. Finally, in [204] the authors take advantage of the MDT measurements to build a Radio Environment Map (REM) by applying spatial interpolation techniques (Bayesian kriging). The REM (Radio Environmental Map) is then used to detect coverage holes and predict the shape of those areas.

#### 4.2.10. Core networks

As we already mentioned in Section 2, the operational aspects of core networks elements can be enhanced through, for example, the automatic configuration of the neighbour cell relations function. In this regard, the idea of applying ML to this function is not new. In [199] the authors study the benefits of using ML to root-cause analysis of session drops, as well as drop prediction for individual sessions. They present an offline Adaboost and SVM method to create a predictor, which is in charge of eliminating/mitigating the session drops by using real LTE data.

#### 4.2.11. Virtualized and softwareized 5G architectures

Currently 5G-PPP projects like SELFNET and key vendors [18] in the industry are working towards the vision of a virtualized and softwareized architecture bringing together the advantages proposed by SDN, NFV and MEC paradigms to introduce programmability and virtualization not only in the cloud, but also at the edge of the network. These novel architectures are expected to be able to support Big Data platforms to collect, manage and analyze all the possible data that can be extracted by the network, to facilitate autonomous decision making. When we go beyond the RAN and we focus on the end-to-end network, ML concepts have already been proposed in different works to build cognitive based techniques to operate the network. An example of these proposals is well summarized by Clark et al. [205]. In this work, a Knowledge Plane is advocated, which would bring many advantages to the networks in terms how the network is operated, automated, optimized and troubleshooted. Conceptually this vision is aligned with different others proposals in other areas, such as the black-box optimization [206], the autonomic self-x architectures [207], or the work



presented in [208]. In this context, the work in [209] analyzes the reasons why the vision proposed in [205] has still not been brought to reality, and the main reason that they find is in the challenges that appear when it comes to autonomously manage a network in a distributed fashion. In particular, the work argues that the emerging trend of centralization in control brought by the novel SDN vision, will significantly reduce this complexity and favour the realization of the ML vision in the network. As a result, in [209] some initial experimental results based on the vision defined in [205] are brought into reality in the context of a SDN based architecture. Further work in this area is carried out in the context of different European H2020 projects [6]. The work in [210] presents a novel cognitive management architecture that manages multiple use cases, like the Service Level Agreement (SLA) and the Mobility Quality Predictor. Both use cases are tackled using machine learning approaches, the Long Short Term Memory, and a per user bandwidth predictor. The work in [211] implements Service Level Agreement (SLA) through ML approaches. It uses an ANN for evaluation of cognitive SLA enforcement of networking services involving Virtualized Network Functions and SDN controllers.

## 5. Challenges for future works

In this section, we focus on some open challenges that still need to be addressed when it comes to making ML based network management a reality.

### 5.1. Big data and privacy

As already mentioned, the ability to handle data from the network, in order to continuously optimize it, will be a fundamental feature of 5G network management, NG SON and future network architectures. In particular, advanced architectures will need to support data platforms sourced from the Big Data concept in order to take full advantage of measurement data, trace data, logs, configuration data, etc., from the entire network. This huge amount of data will need to be handled in order to provide optimal and automatic decision making solutions. Verticals like industrial IoT, smart cities, autonomous cars will generate huge amount of data to be handled through 5G's connectivity with low latency and high data throughput. At the same time, 5G will need to support fast data transfers into the cloud, analytics at the cloud and the edge and in the real time to support mission critical scenarios, public safety, emergency care, etc. For this, traditional ML solutions can be used, or novel approaches based on deep learning, which have been already proven successful when dealing with big amount of data.

This research line, additionally, faces important privacy and confidentiality issues. Big Data raises multiple security concerns. As mentioned in Section 4, data come from different sources of the network, but can also be offered by third parties, e.g., data generated by the user, open data, sensor data, among others. Therefore, to come up with a unified privacy policy is extremely challenging at security and privacy levels, due to the variety and the granularity of the data. If we add to this the speed at which data are created and need to be analysed, the security challenges are huge. In this context, big data is changing the security analytics, where robust and scalable privacy preserving mining algorithms are critical to ensure that the most sensitive private data is secure. As a result, privacy-preserving data mining is a challenging research line that has to be investigated. In particular, in order to guarantee privacy protection, it is important to define the privacy requirements taking into account the lifecycle of the analytics. For example, in the data collection phase, it is important to identify the personal data needed for processing. The idea is to extract only the needed data for the specific purpose. Aggregated information can also be used instead of personal data. In this context, one of the most relevant techniques is anonymization, which is the process of modifying personal data in such a way that no identification is possible. Regarding the data analysis phase, different privacy models are available in the

context of big data analytics, where two of the most important families are: K-anonymity and differential privacy. A review in more detail of the aforementioned methods can be found in [212]. Moreover, in order to protect personal data in databases (data storage phase), *encryption* is a fundamental security technique, which transforms data in a way that only authorized parties can read it. For more information, the reader is referred to [212,213].

### 5.2. Implementation of multi-agent systems for network management

As we have seen in the state of the art sections, reinforcement learning solutions and in particular temporal difference learning, offer an interesting framework for taking online control decisions based on learned experience, while optimizing long term objectives. This is then a useful tool to apply in the context of SON solutions, for the optimization of network functions. When the scenario is limited to a decision maker, or agent, single agent reinforcement learning can be applied, and optimal solutions can be found and demonstrated. However, when in the environments multiple agents are coexisting and interacting among each others, the current approach is to use single agent algorithms and extend them to multi-agent settings. However, this kind of algorithms require a considerable amount of time before finding a solution, which also increases with the state and action spaces. So, reinforcement learning approaches dealing with these issues have to be investigated. Moreover, no proof of convergence is available demonstrating that this approach actually reaches meaningful conclusions. Game theoretic effects also appear, since the policy found by an agent can be affected by the executed policy of the neighbour agent. Even though ML literature offers different algorithms that can find interesting solutions (e.g. NashQ [214]), the space of possible solutions is so big that this kind of approaches is not feasible to be used in a realistic network where the time constraints of RRM/SON problem have to be met. So, more research in the area of multi-agent systems, which are also compatible with real network requirements need to be investigated. But also, research involving cognitive tool kits (knowledge representation, spatial reasoning, acquisition heuristics, consciousness/theory mind among others) and multi-agent systems that rely on reinforcement learning and evolutionary strategies.

### 5.3. Next generation SON for 5G NR

Currently SON only addresses the list of standardized use cases that we have been describing in the survey, and is defined as an optional element. However, to fulfill the objective of a self-organized 5G network, SON has to be designed as an integral part of 5G networks. Telus and Huawei propose in [18] a joint view on the evolution of the NG-SON and propose that in order to provide sustainable network growth and to enable 5G success, NG-SON should be considered as an integral part of the RAN and not just as an add-on feature with a limit set of use cases. NG-SON has to be designed together with NR, the new radio access technology for 5G, which is currently under definition in 3GPP. However, this is not an easy task. It is to expect that the user experience requirements will generate requirements for improved spectrum efficiency and higher data rates, and this will come at the cost of increased network complexity. 5G networks will have to handle an increased number of carriers and aggregated bandwidth, larger frequency range, much higher Multiple-input Multiple-output (MIMO) order, increased number of sites, increased number of RAN features and parameters. This will require increased coordination between the sites and increased number of technologies (in particular LTE/5G). But also, the network architecture must be capable of supporting a hybrid architecture (C-SON and D-SON), a platform to take full advantage of all the data generated in the network, and operators should be able to visualize and monitor the SON adjustment configurations for optimal decision making. Thus, this is an important research line to be considered.

In addition, 3GPP networks will also have to interface other kind of

networks. The autonomous management of multi-technology networks, where heterogeneous networks including different Radio Access Technologies (RAT)s, or different layers of the network are coexisting, e.g., Wi-Fi, mmWave, mobile network layer, transport layer, among others, is still immature. These scenarios will tend to emerge with the advent of the unlicensed spectrum paradigm and with technologies such as LAA or NR. NR, in particular, will be defined to work over a wide range of spectrum opportunities, ranging from sub 6 GHz and up to mmWave spectrum, and under multiple spectrum paradigms, such as licensed, unlicensed and shared. The opportunities of autonomous network management in this area are huge. ML has still not been exploited to handle these networks with intelligence and self-awareness. In particular, the management of densified and heterogeneous, in both technologies and layers, architectures, requires the evolution of NG-SON concepts, which have traditionally been designed and standardized for LTE based networks.

#### 5.4. Network management of novel softwarized and virtualized architectures

The 5G network architecture must be carefully designed to support the NG-SON introduced in previous subsection. First of all, the architecture should be hybrid, in the sense that it has to be able to support both D-SON and C-SON, in order to take full advantage of both paradigms. D-SON shortens the response cycles and C-SON more easily manage large-sized networks. The architecture should also support platforms to collect, and analyze data derived from the network, in order to take full advantage of measurement data, trace data, logs and configurations of the entire network. A platform sourced from Big Data concept should be able to collect as much data as possible, to provide automatic decision making. The architecture should also be programmable in line with the novel SDN paradigm, in order to allow for easy fix and reconfigurability. The platform should also be virtualized in order to allow for cost effective introduction of new features and functions, while meeting the stringent new service verticals' requirements. This infrastructure is to be managed by an orchestrator entity (in coordination with the corresponding virtual network function and virtual infrastructure managers), as proposed in ETSI architecture. The network should not only react to failures, but adapt to the demand, predict it, based on data analytics collected through the network in order to facilitate the network management task. Before reaching this vision, multiple challenges need to be addressed, e.g. the self-coordination problem and the solution of conflicts among SON functions executed in different nodes, or networks, which put the network at risk of instability, or the most appropriate location of SON functions and algorithms, to solve properly the D-SON vs. C-SON implementation issue. Many aspects have to be considered when locating and designing a SON function, e.g. response time, complexity, size of databases, computational capability of nodes, etc. Centralized (i.e. a large number of cells is involved), distributed (approx. 2 cells are involved, coordinating through X2) and local (only one cell is involved) implementations of SON functions have been proposed. No architecture can be claimed superior to the other. The virtualized and softwarized paradigm offers the opportunity to build an elastic SON vision, dynamically moving from one implementation to the other. Research on deep reinforcement learning implementations of the orchestrator will allow it to self-learn after every decision. Automation will require also all the advancements of the Information Technology sector, with increased computational capacity, more CPUs and memory space. However, future orchestrators will need to handle a huge amount of data and learn from them through novel deep learning approaches, the smart network management decisions. This research line is still highly immature and requires a lot of efforts.

## 6. Conclusions

In this work we have motivated the need for ML to be considered as

a crucial and inevitable tool in order to address automation, self-awareness and self-organization in current and future mobile networks. SON capabilities have been included LTE as add on features of LTE. However, the need of automation will be further enhanced with the expected complexity that future 5G network management will have to handle, so that NG-SON needs to be redefined as an embedded feature of 5G networks. On top of that, we have shown that current cellular networks already generate a huge amount of data that if properly stored and managed could bring new insights in how the networks work and offer new challenges for improving network management taking into account the experience that can be gained from these data. We have reviewed the main taxonomy of machine learning and the novel trends that could make this exploitation of data to gain insight of the network a reality. Also, we have discussed open data options, as much as alternatives to get data from the networks, which otherwise are not made available to the academic community. With these motivations in mind, we have started by reviewing the main concepts and taxonomy of SON, network management and ML, and we have reviewed significant academic literature in the area of network management focusing only on solutions based on ML. The work has reviewed the status of this exciting research line, while at the same time highlighting open challenges that we need to deal with in order to make future autonomous network management a reality.

## 7. Competing interests

The authors declare that there is no conflict of interest regarding the publication of this paper.

## Acknowledgment

The research leading to these results has received funding from the Spanish Ministry of Economy and Competitiveness under grant TEC2017-88373-R (Project 5G-REFINE) and by Generalitat de Catalunya grant 2017 SGR 1195. This work also was supported by the Spanish National Science Council and ERFD funds under Project TEC2014-60258-C2-2-R.

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:[10.1016/j.comcom.2018.07.015](https://doi.org/10.1016/j.comcom.2018.07.015).

## References

- [1] Gandalf, Monitoring and self-tuning of RRM parameters in a multi-system network, [Online]. Available:<http://www.celtic-initiative.org/>.
- [2] SOCRATES, Self-Optimisation and self-ConfiguRATION in wirelEss networkS, [Online]. Available:<http://www.fp7-socrates.org/>.
- [3] SEMAFour, Self-Management for Unified Heterogeneous Radio Access Networks, [Online]. Available:<http://fp7-semafour.eu/>.
- [4] SESAME, Small Cells Coordination for Multi-tenancy and Edge services, [Online]. Available:<https://5g-ppp.eu/sesame/>.
- [5] SELFNET, Framework for self-organised network management in virtualised and software defined networks, [Online]. Available:<https://5g-ppp.eu/selfnet/>.
- [6] COGNET, Cognitive Networks, [Online]. Available:<https://5g-ppp.eu/cognet/>.
- [7] O. Aliu, A. Imran, M. Imran, B. Evans, A survey of self organization in future cellular networks, *IEEE Commun. Surv. Tut.* (99) (2012) 1–26.
- [8] Sistelbanda, SN4G SON, [Online]. Available:<http://sistelbanda.es/>.
- [9] Qualcomm, Self managing and enabling seamless roaming, [Online]. Available:<https://www.qualcomm.com/videos/qualcomm-wi-fi-son>.
- [10] Huawei, Huawei's innovative single SON solution, [Online]. Available:<http://www1.huawei.com/en/solutions/broadband-smarter/>.
- [11] A. communications, Powering 4G networks, [Online]. Available:<http://www.airhopcomm.com>.
- [12] Elisa, Elisa automate, mobile word congress 2018, [Online]. Available:<https://elisaautomate.com/>.
- [13] Cellwize, Driving value through SON, [Online]. Available:<http://www.cellwize.com>.
- [14] AVIAT, Wireless products for small cell applications, [Online]. Available:<https://startupgenome.co/aviat-networks>.
- [15] INTRACOM TELECOM, SON automation in Small cell backhaul, INTRACOM

- TELECOM, Mobile Word Congress 2018, [Online]. Available: [http://www.intracom-telecom.com/en/products/wireless\\_network\\_systems/4G\\_smallcell\\_son\\_backhaul/son\\_Automation\\_solutionPaper.htm](http://www.intracom-telecom.com/en/products/wireless_network_systems/4G_smallcell_son_backhaul/son_Automation_solutionPaper.htm).
- [16] C.C.S. (CCS), Ccs metnet self-organising mmwave access and backhaul, [Online]. Available: <https://www.ccs.com/>.
  - [17] S.C. Forum, Small cell industry award, [Online]. Available: <http://www.smallcellforum.org/events/awards>.
  - [18] TELUS, Huawei, Next generation son for 5g, white paper, 2016, [Online]. Available: <http://www.huawei.com/en/industry-insights/outlook/mbb-2020/trends-insights/next-generation-son-for-5g>.
  - [19] 3GPP, Technical Specification Group Services and System Aspects; Telecommunication Management; Self-Organizing Networks (SON); Concepts and requirements (Release 13), TS, 3GPP, 2016.
  - [20] A. Imran, A. Zoha, A. Abu-Dayya, Challenges in 5G: how to empower SON with big data for enabling 5G, *IEEE Netw.* 28 (6) (2014) 27–33.
  - [21] NOKIA, Intelligent self-organizing networks: increased automation for higher network performance with lower costs, White Paper (2013).
  - [22] Ovium, Why fronthaul matters: a key foundation for centralized and cloud ran, white paper, 2015, [Online]. Available: [http://e-blink.com/wp-content/uploads/Why\\_Fronthaul\\_Matters\\_EBLINK\\_Ovium\\_whitepaper.pdf](http://e-blink.com/wp-content/uploads/Why_Fronthaul_Matters_EBLINK_Ovium_whitepaper.pdf).
  - [23] N. Baldo, L. Giupponi, J. Mangues-Bafalluy, Big data empowered self organized networks, 20th IEEE European Wireless, (2014).
  - [24] B. Romanous, N. Bitar, A. Imran, H. Refai, Network densification: challenges and opportunities in enabling 5G, 20th IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD, Special Session: Dense and Elastic Big Data Empowered SON, (2015), pp. 129–134.
  - [25] D. Falko, A study of self-organization mechanisms in ad hoc and sensor networks, *Els. Comput. Commun.* (2008) 3018–3019.
  - [26] PauloValente. Klaine, MuhammadAli. Imran, Oluwakayode Onireti, RichardDemo Souza, A survey on ML techniques applied to self-organizing cellular networks, *IEEE Commun. Surv. Tut.* (2017).
  - [27] D. Kreutz, F.M.V. Ramos, P.E. Verissimo, C.E. Rothenberg, S. Azodolmolky, S. Uhlig, Software-defined networking: a comprehensive survey, *Proc. IEEE* 103 (1) (2015) 14–76, <https://doi.org/10.1109/JPROC.2014.2371999>.
  - [28] 3GPP, Evolved Universal Terrestrial Radio Access Network (EUTRAN); Self configuring and self optimizing network uses case and solutions (Release 9), 3GPP, TR T36.902, v9.2.0, 2010.
  - [29] 3GPP Telecommunication management; Self-Organizing Networks (SON); Concepts and requirements (Release 11), 3GPP, TS 32.500, v11.1.0, 2011.
  - [30] 3GPP, Self Configuration of Network Elements; Concepts and Requirements, Release 9, 3GPP, TS 32.501, 2010.
  - [31] 3GPP, Self-configuration of network elements Integration Reference Point (IRP); Information Service (IS) (Release 10), 3GPP, TS 32.502, v10.1.0, 2010.
  - [32] 3GPP, Self-Configuration of Network Elements Integration Reference Point (IRP); Common Object Request Broker Architecture (CORBA) Solution Set (SS) (Release 8), TR, 3GPP, 2008.
  - [33] Software Management (SWM); Software and Integration Reference Point (IRP) Requirements (Release 10), 3GPP, TR 32.531, 2010.
  - [34] 3GPP, Software management Integration Reference Point (IRP); Information Service (IS) (Release 8), TR, 3GPP, 2009.
  - [35] 3GPP, Software Management (SWM) Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS) (Release 9), Technical Report, 3GPP, 2009.
  - [36] 3GPP, Technical Specification Group Services and System Aspects; Telecommunications management; Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 9), TS, 3GPP, 2009.
  - [37] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) (Release 11), TS, 3GPP, 2014.
  - [38] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP): CORBA solution set (Release 9), TR, 3GPP, 2009.
  - [39] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP): eXtensible Markup Language (XML) definitions (Release 9), TR, 3GPP, 2004.
  - [40] 3GPP, Study of Self-Organizing Networks (SON) related Operations, Administration and Maintenance (OAM) for Home Node B (HNB) (Release 9), TR, 3GPP, 2004.
  - [41] 3GPP, Telecommunication management; Self-Organizing Networks (SON); Study on self-healing (Release 9), TR, 3GPP, 2010.
  - [42] 3GPP, Performance measurements Evolved Universal Terrestrial Radio Access Network (E-UTRAN) (Release 10), TS, 3GPP, 2012.
  - [43] 3GPP, Universal Mobile Telecommunications System (UMTS); UTRAN Iu Interface RANAP Signalling, TS, 3GPP, 1999.
  - [44] 3GPP, Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (Release 12), TS, 3GPP, 2015.
  - [45] 3GPP, LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP) (Release 9), TS, 3GPP, 2010.
  - [46] 3GPP, Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access Network (EUTRAN); X2 Application Protocol (X2AP) (Release 11), TS, 3GPP, 2013.
  - [47] 3GPP, Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS), 3rd Generation Partnership Project (3GPP), (Release 10), TS, 3GPP, 2011.
  - [48] 3GPP, Self-Organizing Networks (SON): Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS); Solution Set (SS) definitions, (Release 10), TS, 3GPP, 2011.
  - [49] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set (SS) definitions, (Release 10), TS, 3GPP, 2011.
  - [50] 3GPP, Telecommunication management; Self-Organizing Networks (SON); Self-healing concepts and requirements (Release 12), TS, 3GPP, 2014.
  - [51] 3GPP, Technical Specification Group Services and System Aspects; Telecommunication Management; Study on Energy Savings Management (ESM) (Release 10), TR, 3GPP, 2010.
  - [52] 3GPP, Technical Specification Group Services and System Aspects; Study on Operations, Administration and Maintenance (OAM) aspects of inter-Radio-Access-Technology (RAT) energy saving (Release 11), TR, 3GPP, 2012.
  - [53] 3GPP, Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Energy Saving Management (ESM); Concepts and requirements (Release 10), TS, 3GPP, 2011.
  - [54] 3GPP, Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol Specification (Release 10), TS, 3GPP, 2010.
  - [55] 3GPP, Performance Management (PM); Performance Measurements UTRAN (Release 9), TS, 3GPP, 2009.
  - [56] 3GPP, Automatic Neighbor Relation (ANR) management, Concepts and Requirements, (Release 11), TR, 3GPP, 2011.
  - [57] 3GPP, Technical Specification Group Services and System Aspects; Telecommunications Management; Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP) (Release 10), TR, 3GPP, 2009.
  - [58] 3GPP, Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP); Network Resource Model (NRM) (Release 9), TS, 3GPP, 2009.
  - [59] 3GPP, TSG SA WG5 (Telecom Management) Meeting 85; Study of implementation alternative for SON coordination, Technical Report, 3GPP, 2012.
  - [60] 3GPP, Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP); Solution Set (Release 10), Technical Report, 3GPP, 2011.
  - [61] 3GPP, Radio Resource Control (RRC); Protocol specification (Release 10), TS, 3GPP, 2011.
  - [62] 3GPP, UTRAN overall description (Release 10), TS, 3GPP, 2011.
  - [63] 3GPP, UTRAN Iu Interface: General Aspects and Principles (Release 10), TS, 3GPP, 2011.
  - [64] 3GPP, UTRAN Iur interface Radio Network Subsystem Application Part (RNSAP) signalling (Release 9), TS, 3GPP, 2010.
  - [65] 3GPP, Study on Network Management (NM) centralized Coverage and Capacity Optimization (CCO) Self-Organizing Networks (SON) function, TS, 3GPP, 2013.
  - [66] 3GPP, Integration Reference Point (IRP) overview and usage guide (Release 10), TS, 3GPP, 2011.
  - [67] 3GPP, Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements (Release 11), TS, 3GPP, 2012.
  - [68] 3GPP, Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) (Release 11), TS, 3GPP, 2014.
  - [69] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS) (Release 11), TS, 3GPP, 2012.
  - [70] 3GPP, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set (SS) definitions (Release 11), TS, 3GPP, 2012.
  - [71] 3GPP, Gap analysis between 3GPP SA5 specifications and NGMN Top Operational Efficiency (OPE) Recommendations (Release 12), TS, 3GPP, 2013.
  - [72] 3GPP, Data formats for multi-vendor plug and play eNodeB connection to the network (Release 12), TS, 3GPP, 2013.
  - [73] 3GPP, Gap analysis between 3GPP SA5 specifications and NGMN Top Operational Efficiency (OPE) Recommendations (Release 12), TR, 3GPP, 2013.
  - [74] 3GPP, Study of heterogeneous networks management (Release 12), TR, 3GPP, 2014.
  - [75] 3GPP, Study on Operations, Administration and Maintenance (OAM) aspects of Network Sharing (Release 12), TR, 3GPP, 2013.
  - [76] 3GPP, Study on Next Generation Self-Optimizing Network (SON) for UTRAN and E-UTRAN (Release 12), TR, 3GPP, 2013.
  - [77] 3GPP, Study on Energy Saving Enhancement for E-UTRAN (Release 12), TR, 3GPP, 2013.
  - [78] 3GPP, Study on Enhancements of OAM aspects of distributed Self-Organizing Networks (SON) functions, Release 12, TR, 3GPP, 2014.
  - [79] 3GPP, Overview of 3gpp release 14.
  - [80] 3GPP, RAN evolution of LTE in release 14, 2016, [Online]. Available: [http://www.3gpp.org/news-events/3gpp-news/1768-ran\\_rel14](http://www.3gpp.org/news-events/3gpp-news/1768-ran_rel14).
  - [81] 3GPP, Study on OAM aspects of SON for AAS-based deployments, TR, 3GPP, 2017.
  - [82] 3GPP, 3rd Generation Partnership Project; Technical Specification Group Radio Access Network; NR; NR and NG-RAN Overall Description; Stage 2 (Release 15), TS, 3GPP, 2018.
  - [83] 3GPP, Technical Specification Group Services and System Aspects; Telecommunication Management; Project scheduling and open issues for SA5, (Release 8), Technical Report, 3GPP, 2010.
  - [84] M. Amirjoo, P. Frenger, F. Gunnarsson, Neighbor cell relation list and physical cell identity self-organization in LTE, ICC Workshops-IEEE International Conference on Communications Workshops, (2008), pp. 31–41.



- [85] F. Parodi, M. Kylvaja, G. Alford, J. Li, J. Pradas, An automatic procedure for neighbor cell list definition in cellular networks, *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, (2007), pp. 1–6.
- [86] 3GPP, Technical Specification Group Radio Access Network, Evolved Universal Terrestrial Radio Access Network (E-UTRAN), Self configuring and Self optimizing Network (SON) Uses Case and Solutions (Release 9), TR, 3GPP, 2011.
- [87] F. Parodi, M. Kylvaja, G. Alford, J. Li, J. Pradas, Recommended practices for multi-vendor SON deployment, D2- NGMN Alliance V1.0, (2014).
- [88] 3GPP, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, TS, 3GPP, 2011.
- [89] NGMN, Recommended practices for multi-vendor SON deployment, NGMN P-SmallCell Work Stream 2 (WS2) first deliverable, (2014).
- [90] 3GPP, Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access E-UTRAN; Overall description; Stage 2, TS, 3GPP, 2015.
- [91] 3GPP, SON Self-healing management, Technical Report, 3GPP, 2011.
- [92] Seppo Hämmäläinen, Henning Sanneck, Cinzia Sartori, LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency, John Wiley and Sons, 2012.
- [93] L.C. Schmelz, M. Amirijoo, A. Eisenblatter, R. Litjens, M. Neuland, J. Turk, A coordination framework for self-organisation in LTE networks, *IEEE International Symposium on Integrated Network Management*, (2011).
- [94] Z. Altman, M. Amirijoo, F. Gunnarsson, H. Hoffmann, I.Z. Kovács, D. Laselva, K. Spaey, A. Tall, H. Van Den Berg, K. Zetterberg, On design principles for self-organizing network functions, *11th International Symposium on Wireless Communications Systems (ISWCS)*, (2014), pp. 454–459.
- [95] 3GPP, Study on Minimization of drive-tests in Next Generation Networks, TR, 3GPP, 2009.
- [96] J. Johansson, W.A. Hapsari, S. Kelley, G. Bodog, Minimization of drive tests in 3GPP (release 11), *IEEE Commun. Mag.* (2012).
- [97] Nokia Siemens Networks, Automating core network management, mobile world congress, 2012, [Online]. Available: <https://networks.nokia.com/solutions/cloud-native-core-network>.
- [98] A. Blenk, A. Basta, M. Reisslein, W. Kellerer, Survey on network virtualization hypervisors for software defined networking, *IEEE Commun. Surv. Tut.* 18 (1) (2016) 655–685.
- [99] A. Valdivieso Caraguay L. Garcia Vilalba, Monitoring and discovery for self-organized network management in virtualized and software defined networks, *Sensors* 17 (2017) 700–731.
- [100] R. Caruana, A. Niculescu-Mizil, An empirical comparison of supervised learning algorithms, *23rd International Conference on Machine Learning*, (2006).
- [101] P. McCullagh, J. Nelder, Generalized Linear Models, Second Edition, Chapman and Hall, Boca Raton, 1989.
- [102] H. Zhang, The optimality of Naive Bayes, *17th International Florida Artificial Intelligence Research Society Conference*, (2004).
- [103] V. Vapnik, An overview of statistical learning theory, *IEEE Trans. Neural Netw.* 10 (5) (1999).
- [104] A.J. Smola, B. Scholkopf, A tutorial on support vector regression, *Stat. Comput.* (2004) 199–222.
- [105] A. Müller, Kernel approximations for efficient SVMs (and other feature extraction methods), *Data Sci. Mach. Learn. Resour.* (2012).
- [106] M.C. Bishop, Pattern Recognition and Machine Learning, Busines Dia, Llc. Springer Science, 2006.
- [107] J.R. Quinlan, Induction of Decision Trees. Machine Learning, Kluwer Academic Publishers, 1986, pp. 81–106.
- [108] L. Rokach, O. Maimon, Data mining with decision trees: theory and applications, *Series in Machine Perception and Artificial Intelligence*, (2008).
- [109] D. Opitz, R. Maclin, Popular ensemble methods: an empirical study, *J. Artif. Intell. Res.* 11 (1999) 169–198.
- [110] L. Rokach, Ensemble-based classifiers, *Artif. Intell.* (2010) 1–39.
- [111] T.G. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting and randomization, *Mach. Learn.* 40 (2000) 139–157.
- [112] T.G. Dietterich, Machine-learning research: four current directions, *Am. Assoc. Artif. Intell.* (1997).
- [113] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and application to boosting, *2nd European Conference on Machine Learning*, (1995), pp. 148–156.
- [114] G. Zoubin, Unsupervised learning, Gatsby Computational Neuroscience Unit University College London, UK, (2004).
- [115] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Section 16.1. Gaussian Mixture Models and k-Means Clustering, *The Art of Scientific Computing*, Third, York: Cambridge University, 2007.
- [116] A. Ultsch, H.P. Simon, Kohonen's self organizing feature maps for exploratory data analysis, *International Neural Network Conference*, (1990).
- [117] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning (2nd ed.), New York: Springer, 2009.
- [118] R. Nock, F. Nielsen, On weighting clustering, *IEEE Trans. Pattern Anal. Mach. Intell.* 28 (2006) 1–13.
- [119] J. Moysen, N. Baldo, L. Giupponi, J. Mangues-Bafalluy, Predicting QoS in LTE hetnets based on location-independent UE measurement, *20th IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks*, Guildford, UK, (2015).
- [120] J. Moysen, L. Giupponi, J. Mangues-Bafalluy, A mobile network planning tool based on data analytics, *Mobile Inf. Syst., Hindawi* (2017).
- [121] S.T. Roweis, K.S. Lawrence, Nonlinear dimensionality reduction by locally linear embedding, *American Association for the Advancement of Science*, (2000), pp. 2323–2326.
- [122] S.T. Roweis, EM algorithms for PCA and SPCA, *Advances in Neural Information Processing Systems*, The MIT Press, 1998.
- [123] V.J. Hodge, J. Austin, A survey of outlier detection methodologies, *Artif. Intell. Rev.* 22 (2004).
- [124] I. de la Bandera, R. Barco, P. Muñoz, I. Serrano, Cell outage detection based on handover statistics, *IEEE Commun. Lett.* 19 (7) (2015).
- [125] P. Muñoz, R. Barco, A. Gómez Andrades, Correlation based time-series analysis for cell degradation, *IEEE Commun. Lett.* (2016).
- [126] S. Schaal, C. Atkeson, Robot juggling: an implementation of memory-based learning, *Control Syst. Mag.* (1994).
- [127] S. Thrun, Learning to play the game of chess, *Advances in Neural Information Processing Systems* 7, 1995, The MIT Press., Cambridge, MA, 1995.
- [128] M.L. Littman, Markov games as a framework for multi-agent reinforcement learning, *11th International Conference on Machine Learning*, (1994), pp. 157–163.
- [129] P. Liviu, L. Sean, Cooperative multi-agent learning: the state of the art, *Auton. Agent Multi Agent Syst.* 11 (3) (2005) 387–434.
- [130] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press, 1998.
- [131] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, *IEEE Trans. Pattern Anal. Mach. Intell.* 35 (8) (2013) 1798–1828.
- [132] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016. <http://www.deeplearningbook.org>.
- [133] 3GPP, Technical Specification Group Services and Systems Aspects; Charging management; Charging Data Record (CDR) parameter description (Release 13), TS, 3GPP, 2016.
- [134] 3GPP, Telecommunication management; Performance Management (PM); Concept and requirements (Release 10), TS, 3GPP, 2010.
- [135] 3GPP, Technical Specification Group TSG RAN; Radio measurement collection for Minimization of Drive Tests (MDT), Release 10, TS, 3GPP, 2010.
- [136] Internet Engineering Task Force (IETF), Information Model for IP Flow Information Export (IPFIX), Technical Report, 3GPP, 2013.
- [137] Orange, The D4D challenge, [Online]. Available: <http://www.d4d.orange.com/en/Accueil>.
- [138] F. Ricciati, P. Widhalm, M. Craglia, F. Pantisano, Estimating population density distribution from network-based mobile phone data, *JRC and AIT*, (2015).
- [139] M. Laner, P. Svoboda, P. Romirer, N. Nikaein, F. Ricciati, A Comparison Between One-Way Delays in Operating HSPA and LTE Networks, *8th Int'l workshop on Wireless Network Measurements (WINMEE'12)*, Pedeborn, Germany, (2012).
- [140] 3GPP, Technical Specification Group Core Network and Terminals; AT command set for User Equipment (UE) (Release 12), TS, 3GPP, 2013.
- [141] OpenCellID, Largest open database of cell towers and geolocation, [Online]. Available: <https://opencellid.org/>.
- [142] OpenSignal, 3G and 4G LTE Cell Coverage Map, [Online]. Available: <https://opensignal.com/>.
- [143] Antenas GSM, [Online]. Available: <http://antenasgsm.com/>.
- [144] Google geolocation api, [Online]. Available: <https://developers.google.com/maps/>.
- [145] N. Bui, J. Widmer, Owl: a reliable online watcher for LTE control channel measurements, *arXiv:1606.00202* (2016).
- [146] H.D. Trinh, N. Bui, J. Widmer, L. Giupponi, P. Dini, Analysis and modeling of mobile traffic using real traces, submitted to *IEEE Personal Indoor and Mobile Radio Communications (PIMRC)* (2017).
- [147] Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), The LENA-EPC network simulator, [Online]. Available: <http://networks.cttc.es/mobile-networks/software-tools/lena/>.
- [148] M. Peng, D. Liang, Y. Wei, J. Li, H.H. Chen, Self-configuration and self-optimization in LTE-advanced heterogeneous networks, *IEEE Commun. Mag.* 51 (5) (2013) 36–45.
- [149] S.S. Mwanje, A. Mitschele-Thiel, Minimizing handover performance degradation due to LTE self organized mobility load balancing, *77th IEEE Conference on Transactions on Vehicular Technology (VTC Spring)*, (2013), pp. 1–5.
- [150] P. Muñoz, R. Barco, I. de la Bandera, Optimization of load balancing using fuzzy Q-Learning for next generation wireless networks, *Expert Syst. Appl., Els.* 40 (2013) 984–994.
- [151] P. Muñoz, R. Barto, I. de la Bandera, Load balancing and handover joint optimization in LTE networks using fuzzy logic reinforcement learning, *Comput. Netw. Els.* (2014) 113–125.
- [152] J. Suga, Y. Kojima, M. Okuda, Centralized mobility load balancing scheme in LTE systems, *8th International Symposium on Wireless Communication Systems*, Aachen, (2011).
- [153] E. Bergner, Unsupervised learning of traffic patterns in self-Optimizing 4th generation mobile networks, *Master of Science Thesis, KTH Computer Science and Communications*, Stockholm, Sweden (2012).
- [154] C.A.S. Franco, J.R.B. de Marca, Load balancing in self-organized heterogeneous LTE networks: a statistical learning approach, *IEEE Latin-American Conference on Communications (LATINCOM)*, (2015), pp. 1–5.
- [155] W. Qin, Y. Teng, M. Song, Y. Zhang, X. Wang, A Q-learning approach for mobility robustness optimization in LTE-SON, *15th IEEE International Conference on, Communication Technology (ICCT)*, (2013), pp. 818–822.
- [156] S.S. Mwanje, A. Mitschele-Thiel, Distributed cooperative Q-learning for mobility-sensitive handover optimization in LTE SON, *IEEE Symposium on the Computers and Communication (ISCC)*, (2014), pp. 1–6.



- [157] P. Muñoz, R. Barco, I. de la Bandera, On the potential of handover parameter optimization for self-organizing networks, *IEEE Trans. Veh. Technol.* (2013).
- [158] N. Sinclair, D. Harle, I.A. Glover, J. Irvine, R.C. Atkinson, Parameter optimization for LTE handover using an advanced SOM algorithm, *IEEE Conference on Transactions on Vehicular Technology (VTC)*, (2013).
- [159] N. Sinclair, D. Harle, I. Glover, J. Irvine, R. Atkinson, An advanced SOM algorithm applied to handover management within LTE, *IEEE Trans. Veh. Technol.* (2013) 183–1894.
- [160] H. Farooq, A. Imran, Spatio-temporal mobility prediction in proactive self-organizing cellular networks, *IEEE Commun. Lett.* 21 (2) (2017) 370–373.
- [161] Z. Ali, N. Baldo, J. Mangues-Bafalluy, L. Giupponi, Machine learning based handover management for improved qoe in LTE, *IEEE/IFIP Network Operations and Management Symposium*, (2016), pp. 794–798.
- [162] E. Ostlin, H.J. Zepernick, H. Suzuki, Macrocell radio wave propagation prediction using an artificial neural network, *IEEE Vehicular Technology Conference (VTC Fall)*, (2004), p. 57.
- [163] A. Quintero, O. García, A profile-based strategy for managing user mobility in third-generation mobile systems, *IEEE Commun. Mag.* (2004) 134–139.
- [164] K. Majumdar, N. Das, Mobile user tracking using a hybrid neural network, *Wireless Networks*, (2005), pp. 275–284.
- [165] R. Razavi, S. Klein, H. Claussen, A fuzzy reinforcement learning approach for self-optimization of coverage in LTE networks, *Bell Labs Tech. J.* 15 (2010) 153–175.
- [166] M. Naseer ul Islam, A. Mitschele-Thiel, A cooperative fuzzy Q-learning for self-organized coverage and capacity optimization, *IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, (2012).
- [167] J. Li, J. Zeng, X. Su, W. Luo, J. Wang, Self-Optimization of coverage and capacity in LTE networks based on central control and decentralized fuzzy Q-learning, *Int. J. Distrib. Sensor Netw.*, Hindawi (2012).
- [168] S. Fan, H. Tian, C. Sengul, Self-optimization of coverage and capacity based on a fuzzy neural network with cooperative reinforcement learning, *J. Wireless Commun. Netw.* (2014).
- [169] A. Galindo-Serrano, L. Giupponi, G. Auer, Distributed learning in multiuser OFDMA femtocell networks, *73rd IEEE Vehicular Technology Conference (VTC Spring)*, (2011), pp. 1–6.
- [170] M. Dirani, Z. Altman, A cooperative reinforcement learning approach for inter-cell interference coordination in OFDMA cellular networks, *8th Int Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, (2010), pp. 170–176.
- [171] M. Bennis, S.M. Perlaza, P. Blasco, Self-Organization in small cell networks: a reinforcement learning approach, *IEEE Trans. Wireless Commun.* 12 (7) (2013).
- [172] M. Simsek, A. Czylik, A. Galindo-Serrano, L. Giupponi, Improved decentralized Q-learning algorithm for interference reduction in LTE-femtocells, *5th International Workshop on Self-Organizing Networks IWSON*, Glasgow, (2015).
- [173] M. Miozzo, L. Giupponi, M. Rossi, P. Dini, Distributed Q-learning for energy harvesting heterogeneous networks, *IEEE ICC 2015 Workshop on Green Communications and Networks with Energy Harvesting, Smart Grids, and Renewable Energies*, (2015).
- [174] A. Dudnikova, P. Dini, L. Giupponi, D. Panno, Multiple criteria plus fuzzy logic switch off method for dense heterogeneous networks, *20th IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks*, Guildford (UK), (2015).
- [175] E. Terner, P. Agyapong, L. Hu, A. Dekorsy, Energy savings in heterogeneous networks with clustered small cell deployments, *11th IEEE International Symposium on Wireless Communications Systems (ISWCS)*, (2014).
- [176] E. Terner, P. Agyapong, L. Hu, A. Dekorsy, Database-aided energy savings in next generation dual connectivity heterogeneous networks, *IEEE WCNC'14 Track 3: Mobile and Wireless Networks*, (2014).
- [177] J. Moysen, L. Giupponi, A reinforcement learning based solution for self-healing in LTE networks, *80th IEEE Vehicular Technology Conference (VTC Fall)*, Vancouver, Canada, (2014).
- [178] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. Ali Imran, A. Abu-Dayya, A cell outage management framework for dense heterogeneous networks, *IEEE Trans. Veh. Technol.* 64 (2015) 2097–2113.
- [179] A. Saeed, O.G. Aliu, M.A. Imran, Controlling self healing cellular networks using fuzzy logic, *IEEE Wireless Communications and Networking Conference (WCNC)*, Paris, France, (2012).
- [180] F. Chernogorov, J. Turkka, T. Ristaniemi, A. Averbuch, Detection of sleeping cells in LTE networks using diffusion maps, *IEEE 73rd Vehicular Technology Conference (VTC Spring)*, (2011), pp. 1–5.
- [181] E.J. Khatib, R. Barco, A. Gomez-Andrades, I. Serrano, Diagnosis based on genetic fuzzy algorithms for LTE self-Healing, *IEEE Trans. Veh. Technol.* (2015).
- [182] R.M. Khanafer, B. Solana, J. Triola, R. Barco, L. Moltsen, Z. Altman, P. Lazaro, Automated diagnosis for UMTS networks using bayesian network approach, *IEEE Trans. Veh. Technol.* 57 (2008).
- [183] G.F. Ciocarlie, S.N. Sanneckaczki, Detecting anomalies in cellular networks using an ensemble method, *9th CNSM and Workshops*, (2013).
- [184] A. Multazamah, S. Navrati, R. Abhishek, Efficient cell outage detection in 5G het-nets using hidden Markov model, *IEEE Commun. Lett.* 20 (3) (2016).
- [185] W. Xue, M. Peng, Y. Ma, H. Zhang, Classification-based approach for cell outage detection in self-healing heterogeneous networks, *IEEE Wireless Communications and Networking Conference (WCNC)*, (2014), pp. 2822–2826.
- [186] A. Zoha, A. Saeed, A. Imran, M.A. Imran, A. Abu-Dayya, A SON solution for sleeping cell detection using low-dimensional embedding of mdt measurements, *IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, (2014), pp. 1626–1630.
- [187] S. Chernov, F. Chernogorov, D. Petrov, T. Ristaniemi, Data mining framework for random access failure detection in LTE networks, *25th IEEE Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*, (2014), pp. 1321–1326.
- [188] R. Barco, V. Wille, L. Diez, System for automated diagnosis in cellular networks based on performance indicators, *Eur. Trans. Telecommun.* (2005) 399–409.
- [189] R.M. Khanafer, B. Solana, J. Triola, R. Barco, L. Moltsen, Z. Altman, P. Lazaro, Automated diagnosis for UMTS networks using Bayesian network approach, *IEEE Trans. Veh. Technol.* 57 (2008) 2451–2461.
- [190] H.Y. Lateef, A. Imran, A. Abu-dayya, A framework for classification of self-organising network conflicts and coordination algorithms, *IEEE Personal Indoor and Mobile Radio Communications PIMRC*, (2013).
- [191] O. Iacoboaia, B. Sayrac, S. Jemaa, P. Bianchi, SON coordination for parameter conflict resolution: a reinforcement learning framework, *IEEE Wireless Communications and Networking Conference 2014 (IEEE WCNC 2014)*, (2014).
- [192] P. Muñoz, R. Barco, S. Flores, Conflict resolution between load balancing and handover optimization in LTE networks, *IEEE Commun. Lett.* 18 (2014) 1795–1798.
- [193] J. Moysen, L. Giupponi, Self-coordination of parameter conflicts in D-SON architectures: a markov decision process framework, *EURASIP J. Wirel. Commun. Netw.* (2015).
- [194] J. Moysen, M. García Lozano, L. Giupponi, S. Ruiz, Conflict resolution in mobile networks: a self-coordination framework based on non-dominated solutions and machine learning for data analytics, *IEEE Comput. Intell. Mag.* 13 (2018) 52–64.
- [195] F. Chernogorov, T. Nihtilä, QoS verification for minimization of drive tests in LTE networks, *75th IEEE Vehicular Technology Conference (VTC Spring)*, (2012), pp. 6–9.
- [196] F. Chernogorov, J. Puttonen, User satisfaction classification for minimization of drive tests QoS verification, *24th IEEE Personal Indoor and Mobile Radio Communications (PIMRC)*, London, United Kingdom, (2013), pp. 2165–2169.
- [197] J. Moysen, L. Giupponi, J. Mangues-Bafalluy, On the potential of ensemble regression techniques for future mobile network planning, *21th IEEE Symposium on Computers and Communications (ISCC)*, Messina, Italy, (2016).
- [198] J. Moysen, L. Giupponi, J. Mangues-Bafalluy, A machine learning enabled network planning tool, *27th IEEE Personal Indoor and Mobile Radio Communications (PIMRC)*, Valencia, Spain, (2016).
- [199] B. Daróczy, A. Benczúr, P. Vaderna, Machine learning based session drop prediction in LTE networks and its SON aspects, *5th International Workshop on Self-Organizing Networks (IWSON)*, Glasgow, (2015).
- [200] J. Turkka, D. Gil, Anomaly detection framework for tracing problems in radio networks, *10th International Conference on Networks (ICN)*, (2011).
- [201] R. Romeikat, B. Bauer, T. Bandh, G. Carle, H. Sanneck, L.C. Schmelz, Policy-driven workflows for mobile network management automation, *6th International Wireless Communications and Mobile Computing Conference (IWCMC)*, (2010), pp. 1111–1115.
- [202] V. Buenestado, M. Toril, S. Luna, J.M. Ruiz, A. Mendo, Self-tuning of remote electrical tilts based on call traces for coverage and capacity optimization in LTE, *IEEE Trans. Veh. Technol.* (2017).
- [203] M. Toril, R. Acedo-Hernández, S. Luna-Ramírez, A. Sánchez, C. Úbeda, Automatic clustering algorithms for indoor site selection in LTE, *Hindawi Mobile Inf. Syst.* (2017).
- [204] A. Galindo-Serrano, B. Sayrac, S. Ben Jemaa, J. Riihijärvi, P. Mähönen, Harvesting MDT data: radio environment maps for coverage analysis in cellular networks, *International Conference on Cognitive Radio Oriented Wireless Networks (CROWNCOM)*, (2013).
- [205] D. Clark, et al., A knowledge plane for the internet, *Proc. of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, (2003).
- [206] L.M. Rios, N.V. Sahinidis, B. Sayrac, Derivative free optimization: a review of algorithms and comparison of software implementations, *J. Global Optim.* (2013) 1247–1293.
- [207] H. Derbel, N. Agoulmine, M. Salan, Anema: autonomic network architecture to support self-configuration and self-optimization in ip networks, *Comput. Networks* (2009) 418–430.
- [208] e.a.M. Zorzi, COBANETS: a new paradigm for cognitive communications systems, *International Conference on Computing, Networking and Communications ICNC*, (2016).
- [209] A. Mestres, et al., Knowledge defined networking (2016) arXiv:1606.06222.v2.
- [210] I. Ben Yahia, et al., Cognitive 5G networks: comprehensive operator use cases with machine learning for management operations, *20th Conference on Innovations in Clouds, Internet and Networks (ICIN)*, (2017).
- [211] J. Bendriss, et al., AI for SLA management in programmable networks, *International Conference on Design of Reliable Communication Networks*, (2017).
- [212] European Union Agency for Network and Information Security (enisa), Privacy by design in big data, An overview of privacy enhancing technologies in the era of big data analytics (2015).
- [213] K.A. Sallehab, L. Janczewski, Technological, organizational and environmental security and privacy issues of big data: a literature review, *Procedia Computer Science*, Elsevier, 2016, pp. 19–28.
- [214] J. Hu, M.P. Wellman, Nash Q-learning for general-sum stochastic games, *J. Mach. Learn. Res.* 4 (2003) 1039–1069.