Autonomic Mobile Virtual Network Operators for Future Generation Networks

Fabrizio Granelli *Senior Member, IEEE*, and Riccardo Bassoli, *Member, IEEE*

Abstract

The effort of telecommunications operators in 5G design and implementation is oriented to effective and efficient verticals support. Full network softwarization will deploy tools such as software-defined networks and network function virtualisation to allow dynamic service provisioning on the same physical infrastructure. However, current 5G proposed architecture still require further enhancement to guarantee full real-time on-demand services. Current virtualisation technologies still miss complete automation and adaptation, while they do not have dynamic negotiations capabilities. These issues are bounding the desired complete automation and the significant reduction in OpEx and CapEx. Thus, the article propose a full architecture to realise Autonomic Mobile Virtual Network Operators. This autonomic virtual operators can be deployed by Internet Service Providers to guarantee efficient and effective network adaptation to unexpected events and real-time resource requests. The objective of an AMVNO is to exploit softwarization of networks and experiential intelligence to play the role of a local operator by performing network management and service release without human intervention. Its autonomic character can also provide proactive actions against unwanted network states.

Index Terms

5G, virtualisation, software-defined networking, autonomic mobile virtual network operator.

I. Introduction

In the last years, the vision of future generation networks (5G and beyond) revealed a relevant broadening of its scope with respect to the previous generations. 5G and beyond 5G (B5G) networks will need to address a significant heterogeneity not only of the involved technologies but also of the various kinds of services, which will be offered to the customers of internet service providers (ISPs) and mobile virtual network operators (MVNOs).

F. Granelli and R. Bassoli are with the Department of Information Engineering and Computer Science, at the University of Trento, Trento, Italy (e-mail: {fabrizio.granelli, riccardo.bassoli}@unitn.it).

Indeed, telecommunications' operators are now strongly focusing on 'verticals' to increase both their customers and revenues in future 5G and B5G networks. The main method to achieve that is the gradual replacement of current dedicated network equipment with general purpose one. In fact, this will be the means to support a full network softwarization process to guarantee the desired flexibility to upgrade and to reconfigure frequently the network infrastructure at acceptable costs. The main enablers identified by research community and industry to realise full network softwarization and virtualisation are software-defined networking (SDN) and network function virtualisation (NFV). Especially, these technologies will also allow slicing of network infrastructure and resources. In this way, future infrastructure providers (InPs) will be able to support the various requirements of different ISPs and MVNOs with the same physical infrastructure. The leading role of current mobile operators as future InPs is also underlined by the growing diffusion of MVNOs. According to a report of GSMA Intelligence, the number of MVNOs increased 70% worldwide in the period 2010-2015, achieving the quota of 1017 in June 2015.

Figure 1 depicts a possible B5G scenario. Different kinds of end-users will simultaneously require several types of services, with specific demands. The network will consist of three macro areas: random access technologies/random access networks (RATs/RANs), edge and core networks. The RATs/RANs will provide access via multiple wireless technologies (e.g. cellular, wireless local area network (WLAN), satellite, etc.). In parallel, the edge/core networks will have nodes such as SDN switches and datacentres of different sizes, which will host virtual network functions (VNFs) and services. Eventually, the core network will also be able to support satellite links and satellite-based SDN switches.

Nevertheless, the current status of expected 5G and B5G architectures still lacks optimisation to perform full real-time on-demand operations in such heterogeneous environment. Next, present virtualisation technologies still require a lot of manual configuration and intervention, which are not suitable to guarantee effective management of future low-latency (1-10ms) end-to-end services. Current SDN/NFV based systems do not integrate dynamic negotiations capabilities, and are not fully self-adaptive and self-driven. These aspects significantly limit the desired reduction in operational expenditure (OpEx) and capital expenditure (CapEx), and the dreamed real automation.

During the last decade, a growing research trend focused on the extensive deployment of cognition to make fully-autonomous cognitive networks. Especially, in the last few years, the research community started the study of self-organised networks (SONs) [1] in parallel to virtualisation, by applying machine learning and cognitive algorithms mainly towards self-healing and self-management. This increasing tendency has resulted in the ongoing standardisation work of new European Telecommunications Standards Institute (ETSI) group called Experiential Network Intelligence (ENI). Its purpose is to define a cognitive network management architecture to improve operators' experience with artificial intelligence.

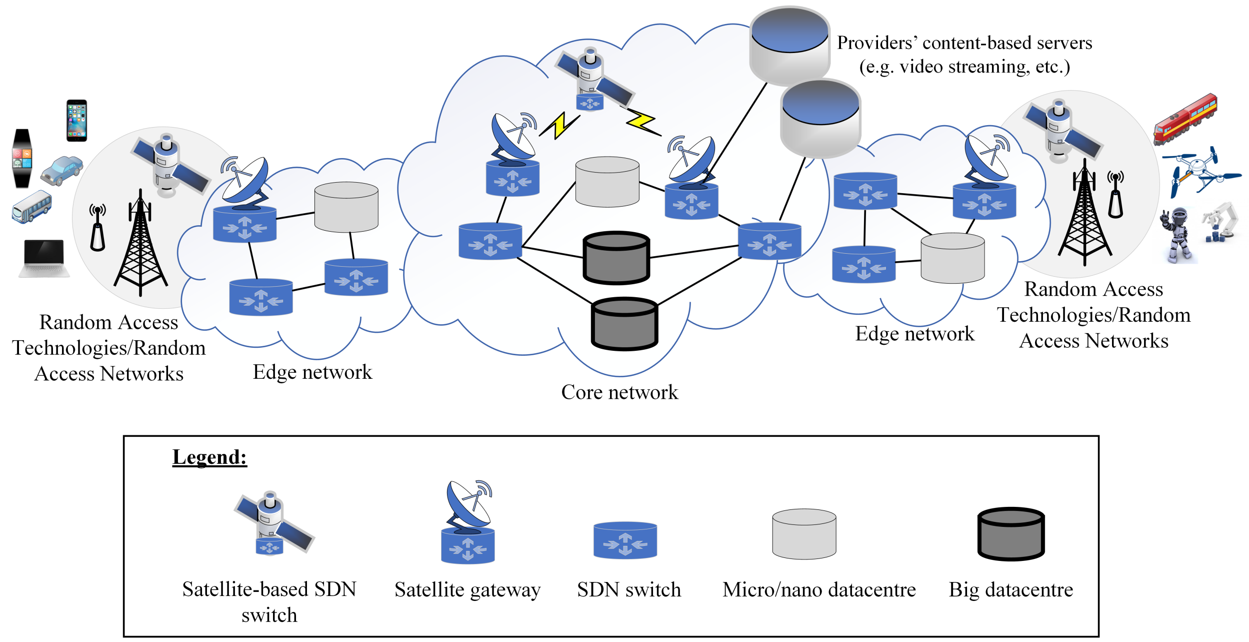

Providers' content-based servers
(e.g. video streaming, etc.)
Random Access Technologies/Random Access Networks
Edge network
Core network
Edge network
Random Access Technologies/Random Access Networks
Legend:
Satellite-based SDN switch
Satellite gateway
SDN switch
Micro/nano datacentre
Big datacentre

Fig. 1. Possible future generation networks scenario. The majority of the physical network infrastructure will include nodes like SDN switches, 'micro/nano centres' (with limited storage, processing and communication capabilities) and big datacentres (with very high storage, processing and communication capabilities). The network will be divided into three main areas: the RATs/RANs, the edge network and the core network. The RATs/RANs will include heterogeneous wireless access technologies such as current 4G base stations, WLAN access points, mmWave base stations and satellites. Moreover, the core network will merge both terrestrial and satellite infrastructures, given the presence of satellite gateways and satellite-based SDN switches. The future network infrastructure will support a great variety of end-to-end services, including mobile broadband, vehicular communications, public transports, unmanned vehicles, automated factories (machine-to-machine and human-to-machine) and smart cities. End-to-end communications will possibly involve two RANs or a RAN and servers, located in the core/edge network. In the figure, 'clouds' overlapping means the areas of the network are interconnected while it is not clearly defined the border between edge and core.

Side by side, in 2010, the concept of Cognitive Mobile Virtual Network Operator (CMVNO) [2] raised next to the mainstream investigation about the cognitive management of networks from 'physical' mobile operators' perspective. The main aim of CMVNOs was the enhancement of commercial and business possibilities of both ISPs and newcomers to the market. However, the deployment of CMVNOs was limited to RANs in order to enhance spectrum utilisation and dynamic spectrum leasing.

Given the strong standardisation and research effort towards cognitive/autonomic networks, it is important to extend the idea of CMVNOs not only to physical resources but also to the management of entire end-to-end virtualised networks such as virtual slices embracing RAT/RAN, edge and core network. That because in the context of 5G/B5G, network performance is oriented to end-to-end services and not to optimisation of specific network characteristics.

This article aims to describe the structure, architecture and protocols to realise an Autonomic Mobile Virtual Network Operator (AMVNO). The scope of an AMVNO is to optimise autonomously and dynamically resource slicing, routing processes and VNFs placement. The realisation of AMVNOs can allow ISPs and virtual operators to rent parts of InPs' physical networks for efficient end-to-end service provisioning. Moreover, InPs' can also delegate to virtual operators the management and maintenance of sections of their network. Such autonomic virtual operators would be required to deploy unsupervised machine learning approaches for effective real-time self-management, self-adaptation and self-healing.

The AMVNO environment relies on the full integration of SDN and NFV in a unique architecture, which is the current objective for a significant part of the research community. Nevertheless, this article studies the novel concept of AMVNOs, which requires the efficient integration of autonomic operations into SDN-NFV-based architectures. The autonomic network operations are possible via the use of an autonomic control plane, mainly including a cognitive hypervisor. This hypervisor uses specific interfaces and procedures for resource reservation/assignment and for management of the virtualised infrastructure.

AMVNOs must guarantee network performances on the basis of specific real-time resource requests by the end-to-end services and they should be capable to respond effectively to sudden unexpected events without human intervention. The AMVNO can exploit experiential intelligence not only for a posteriori adaptation (reactive) but also to predict and to anticipate potential unwanted network states (proactive). This article on AMVNOs investigates the possibility of a network management strategy independent of human intervention, while also describing some important performance metrics and design/implementation guidelines.

II. Background on Fundamental Concepts

A. Autonomics, Virtualisation and Slicing

The idea of 'autonomic system' was firstly presented by [3] in 2003. Since then, cognition and autonomics have become more and more interesting in the context of telecommunications. In general, *cognitive systems* [1] are systems, which employ machine learning algorithms for self-configuration, self-optimisation and self-healing. These learning algorithms can belong to three main categories: supervised learning, unsupervised learning and reinforcement learning. The first type requires human supervision for training and monitoring, the second one does not need any human supervision, while the third one is based on rewards/penalties after decisions are made. On the other hand, *autonomic systems* are specific unsupervised cognitive systems, which completely manage themselves given just high-level preset objectives.

While several cognitive approaches have been developed in previous and current generations of networks [1], the realisation of autonomic systems inside networks needed a flexible/reconfigurable control infrastructure to find its fertile ground. That happened with the raise of the concepts of virtualisation and 'softwarization' of networks, via the diffusion of SDN and NFV paradigms.

Software-defined networking [4] is a technology to allow software-based control of data-paths and routing strategies of networks. The main idea behind such paradigm is the detachment of control and data planes. The central entity of SDN architecture is the SDN controller, which updates the routing tables and routing policies at the network nodes (SDN switches). An application layer is provided to place the different networking applications such as data path control, user authentication and mobility management. There are different protocol and architectures proposed by standardisation bodies and industry; however, the standard protocol currently used for communications between controller and network devices is Openflow.

The aim of Openflow secure protocol is to change the flow tables at the so called Openflow SDN switches. These flow tables contain the rules, actions and policies to handle data traffic. Moreover, SDN switches collect statistics to improve network analysis by SDN controller. A software change in the SDN controller will modify the behaviour of the whole network.

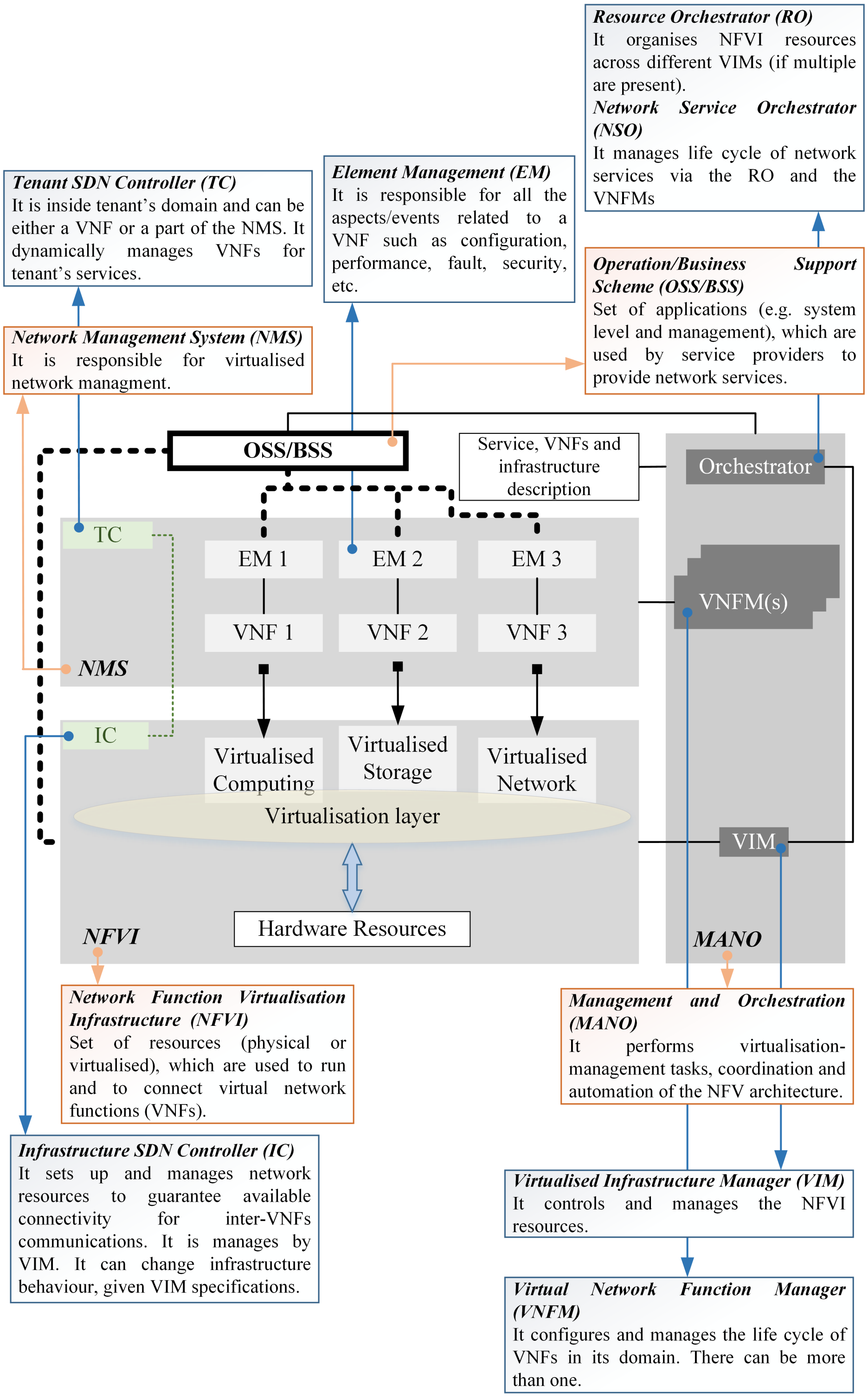
Network function virtualisation mainly aims to decouple network functions from network hardware, to provide flexible deployment of network functions and dynamic scaling. As SDN systems, NFV logic architecture is composed of three layers. The first layer hosts the physical resources (computing, storage and network). The second layer performs management and orchestration of the virtual resources, by translating needs of services into actual deployment of network functions. Finally, the third layer contains the software-based services, which are the virtual network functions to be used. In NFV architectures, multiple control protocols [5] are available.

Though SDN and NFV were born independently, part of research effort during the last years focused on the design of a common SDN-NFV system. Nevertheless, the definition of a unique SDN-NFV architecture is challenging and it is still an open issue due to the complex interactions among them [7].

Three main approaches for a unified SDN-NFV architecture can be identified in the literature.

The first one (Figure 2), proposed by the ETSI working group [8], is based on the majority of research works. This framework is composed by four main macro-blocks

- Operation/Business Support Scheme (OSS/BSS): the set of applications used by ISPs to provide their services;
- Network Management System (NMS), which manages the virtualised network. It contains VNFs and their respective element managements (EMs), and the tenant SDN controller (TC);



Resource Orchestrator (RO) It organises NFVI resources across different VIMs (if multiple are present).

Network Service Orchestrator (NSO) It manages life cycle of network services via the RO and the VNFMs

Tenant SDN Controller (TC) It is inside tenant's domain and can be either a VNF or a part of the NMS. It dynamically manages VNFs for tenant's services.

Element Management (EM) It is responsible for all the aspects/events related to a VNF such as configuration, performance, fault, security, etc.

Network Management System (NMS) It is responsible for virtualised network managment.

Operation/Business Support Scheme (OSS/BSS) Set of applications (e.g. system level and management), which are used by service providers to provide network services.

Network Function Virtualisation Infrastructure (NFVI) Set of resources (physical or virtualised), which are used to run and to connect virtual network functions (VNFs).

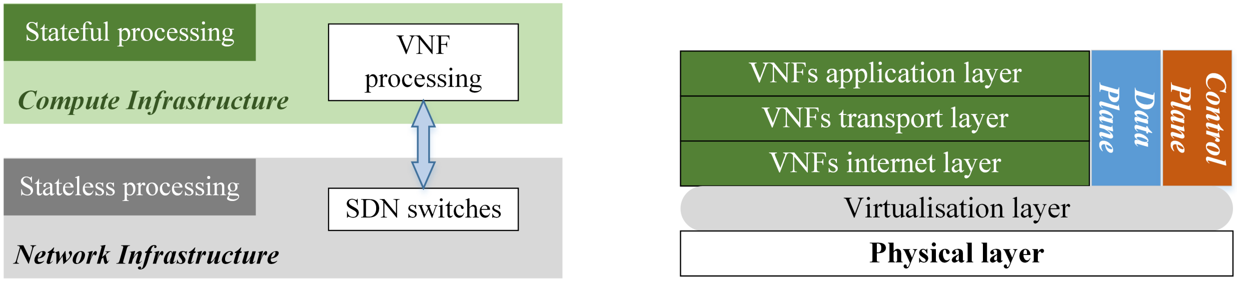
Management and Orchestration (MANO) It performs virtualisation-management tasks, coordination and automation of the NFV architecture.

Infrastructure SDN Controller (IC) It sets up and manages network resources to guarantee available connectivity for inter-VNFs communications. It is manages by VIM. It can change infrastructure behaviour, given VIM specifications.

Virtualised Infrastructure Manager (VIM) It controls and manages the NFVI resources.

Virtual Network Function Manager (VNFM) It configures and manages the life cycle of VNFs in its domain. There can be more than one.

Fig. 2. Architecture proposed by ETSI working group. Four macro-blocks compose this framework, called OSS/BSS, NMS, NFVI and MANO: their role is described in the respective red rectangles. On the other hand, the functions of internal blocks are summarised in the respective blue rectangles.



(a)

(b)

Fig. 3. (a) SDN-NFV architecture proposed by [6]. It consists of two main layers on top of the physical infrastructure. The first layer, performing stateless processing, mainly includes SDN switches by providing connectivity. The second layer, performing stateful processing, embraces VNFs and their functions (b) SDN-NFV architecture proposed by [7]. The pile has a horizontal and a vertical orientation. The former is referred to VNFs, which can perform functions belonging to layers from internet up to application. The latter is focused on management and connectivity for both data and control planes, via SDN.

- Network Function Virtualisation Infrastructure (NFVI), which includes physical and virtualised resources to run the VNFs, and the infrastructure SDN controller (IC);
- Management and Orchestration (MANO), which performs management, coordination and automation of NFV architecture. This functionality groups the VNF manager (VNFM), the virtualised infrastructure manager (VIM) and the orchestrator.

As an alternative, Figure 3(a) depicts the architecture proposed by [6]. This framework has two layers: network and compute infrastructures. The former is an SDN infrastructure, which provides connectivity enhanced by the additional support of VNFs processing. The latter is responsible for VNFs hosting and processing.

Finally, Figure 3(b) shows the architecture proposed by [7]. It is a model with two orientations: a vertical one represents the stack of VNFs for different layers, while a horizontal one includes data and control planes for SDN-based management of all the virtualised layers.

Network slicing [9] is a technique based on virtualisation to create subsets of a network. Such separate subsets can be used to improve resource management and flexibility or for resource allocation to different tenants. There are mainly two ways to slice: Quality-of-Service (QoS) slicing (to guarantee specific network performance requirements to specific users), and infrastructure sharing slicing (for physical network compartmentalisation). Another classification can be according to the scope of slicing: spectrum-level slicing, infrastructure-level slicing and network-level slicing.

Slicing procedures can enhance network adaptability, providing dynamic network management according to operators' policies. Its main enablers are SDN and NFV. Normally, the hardest portion of network to slice is the base station due to the high performance variability of wireless access links and fluctuating capacity.

B. Resource Reservation

Resource reservation procedures can play a key role in network virtualisation. They can let optimal assignment/mapping of physical network resources by additionally deploying effective traffic engineering techniques. Resource reservation represents a 'cross-domain' element since it plays a role in both virtualised and physical domains.

Regarding radio resource reservation in heterogeneous networks, IEEE 1900.4 can be an effective solution for resource reservation. Particularly, its proposed framework provides a suitable architecture for efficient management of terminal, RAN and network reconfigurations.

Resource Reservation Protocol (RSVP) is a well-known reservation protocol with the aim of guaranteeing limited latency to delay-sensitive applications. The protocol can reserve resources for traffic flows from source to sink at all the intermediate nodes.

RSVP is a receiver-oriented reservation protocol, where both data and control packets are sent on the same identified route.

The extension called RSVP Traffic Engineering (RSVP-TE) (see *RFC 5151*) introduced various objects for traffic engineering and additional features such as fast reroute extension. Especially, this extension was proposed to rapidly redirect traffic (less than 100ms) in case of network issues. The idea behind that was the establishment of backup paths to repair locally the tunnels created for data flows.

Next, Section V will show the role of resource reservation in a practical use case of AMVNO.

III. AMVNO: Vision and Design Guidelines

This section describes the proposed architecture for an AMVNO. An AMVNO merges the SDN-NFV-based architecture with autonomics and resource management protocols to create a unique system, which can provide unsupervised and intelligent service towards optimal end-to-end QoS provisioning.

Figure 4 shows how the autonomic control plane can be integrated in SDN-NFV architectures (see Subsection II-A). The maximisation of effectiveness of unsupervised learning algorithms forces to think about autonomics as a parallel control plane and not only as a block within existing architectures. That implies the extension of legacy SDN-NFV architectures towards an additional third dimension. As a consequence, another type of logical interconnections are defined, which represents inter-plane logical interactions. The absence of specific blocks at the end of pins at autonomic control plane means that the logical structure of this plane strongly depends on its unsupervised learning and data mining algorithms.

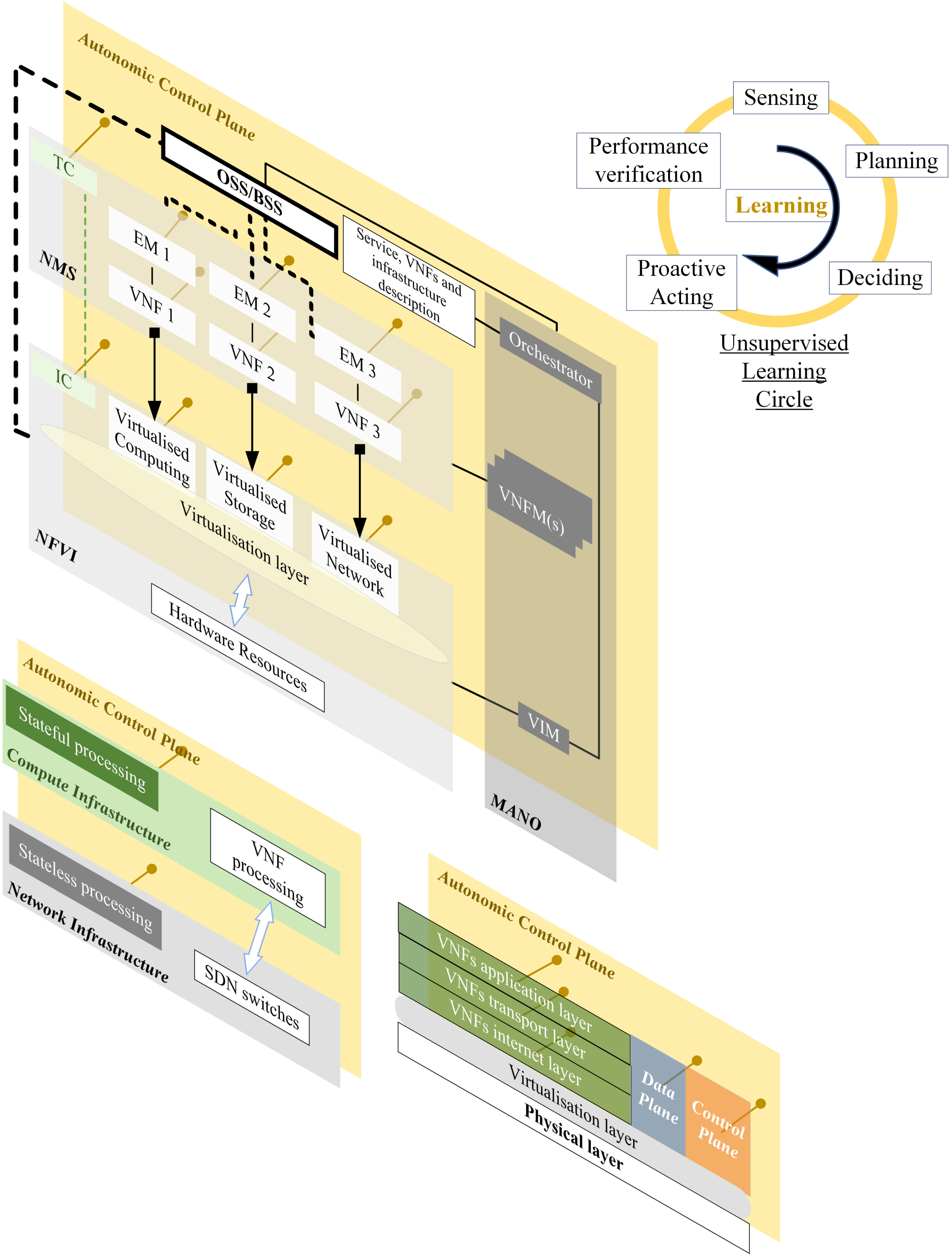


Fig. 4. Placement and role of the autonomic control plane, referring to previously presented SDN-NFV architectures. Pins represent the logical interfaces/connections through which the autonomic control plane can gather information (data mining) and can perform procedures of the unsupervised learning circle.

That also significantly affects how information from all the blocks in the foreground plane is gathered, processed and used.

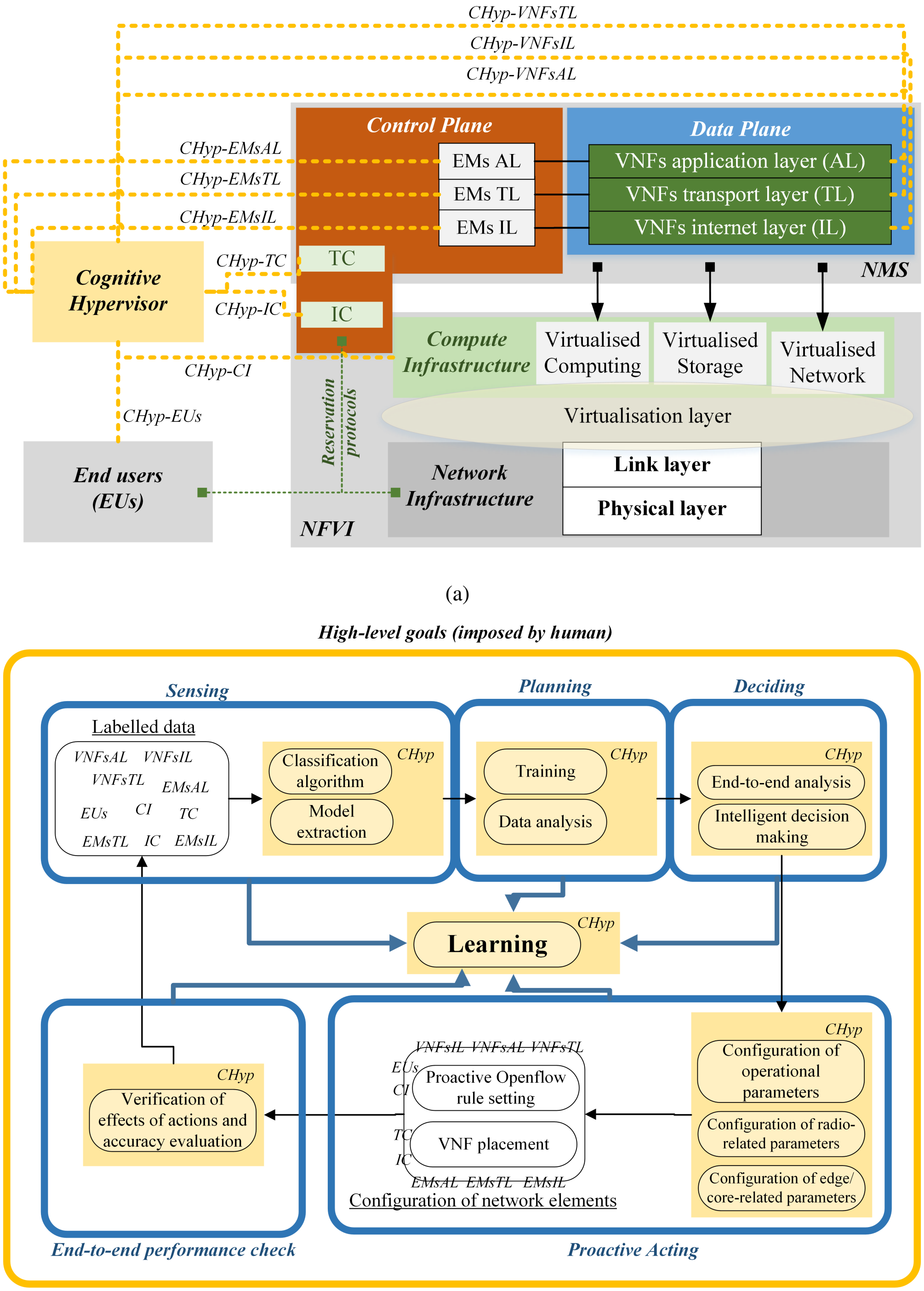
The AMVNO with the architecture of Figure 4 takes advantage of a ’cross-layer/cross-blocks’ knowledge, as it needs knowledge from both TC and IC, from VNFs and EMs (ETSI architecture in Figure 2). Additionally, an AMVNO exploits the stateless and stateful information from both SDN switches and VNFs (architecture in Figure 3(a)). Finally, it requires strict interaction with both SDN and NFV, which spans from internet layer to application layer (architecture in Figure 3(b)). The autonomic control plane learns by performing a sequence of operations called *sensing*, *planning*, *deciding*, *proactive acting* and *performance verification*. Especially, these represent the actions of an *unsupervised learning circle*, which has no human in its loop.

Figure 5(a) depicts the functional-logical architecture of an AMVNO. In this figure, we have drawn an SDN-NFV architecture, which includes and merges all the legacy characteristics that have been separately described by Figure 2 and Figure 3.

The novel *cognitive hypervisor* performs all the functionalities of autonomic control plane and it replaces MANO and OSS/BSS blocks of Figure 2. In particular, it knows the infrastructure’s description and manages all the available services, system-level/management applications and VNFs. Cognitive hypervisor’s interfaces can mainly perform *managerial* and *informative functions*:

- *CHyp-TC* and *CHyp-IC* connect the cognitive hypervisor with TC and IC respectively. They are both informative and managerial since they are used for data gathering (from controllers) and management of SDN controllers’ operations (i.e. SDN control plane management).
- *CHyp-EUs* connects the cognitive hypervisor with end users. It is mainly employed for users’ data gathering.
- *CHyp-EMsAL*, *CHyp-EMsTL* and *CHyp-EMsIL* connect the cognitive hypervisor with EMs at all layers. They are used to gather EMs’ control data and control VNFs’ information. Next, they can also be employed to provide commands to EMs, to manage VNFs’ control plane behaviour.
- *CHyp-VNFsAL*, *CHyp-VNFsTL* and *CHyp-VNFsIL* connect the cognitive hypervisor with VNFs at all layers. They are mainly used to collect information from VNFs’ data plane conditions and performance.
- *CHyp-CI* connects the cognitive hypervisor with the compute infrastructure. Its main aim is to get information about the status of computing/storage/network virtualised resources.

Next, reservation protocols/techniques represent bridges between SDN-NFV control plane and physical network resources (located at either end-users or network infrastructure). Traffic engineering and advanced characteristics of such protocols can improve data gathering, resource management and response efficiency



(a)

(b)

Fig. 5. (a) Functional-logical architecture of an AMVNO (b) Schematic representation of the internal-logical structure of the cognitive hypervisor.

to network changes.

Figure 5(b) shows the internal-logical structure and operations of the cognitive hypervisor. A priori high-level goals and requirements are defined by network managers or infrastructure owners: such parameters/laws represent precise boundaries, the cognitive hypervisor cannot cross via its decisions/actions. Such restrictions can be used to implement policies or regulations the system has to follow.

The cognitive hypervisor's operations are organised into five main phases:

1) *Sensing* – Given specific objectives, the logical entities in Figure 5(a) provide labelled data to the hypervisor, using mining algorithms. Next, this information is internally processed by classification algorithms. Finally, models and patterns are inferred to improve future data collection and behavioural prediction.
2) *Planning* – Data and patterns of Phase 1 are fundamentals to train the machine learning algorithm of the hypervisor. Then, data are analysed and prepared for the next phase.
3) *Deciding* – Classified and processed information is analysed from an end-to-end perspective to make a decision on how to address specific network issues and negative states.
4) *Proactive acting* – Parameters, which are referred to both virtual network operations and network resource configuration (RAN/edge/core), are set up. This action is proactively performed: this means the cognitive hypervisor is always predicting network future states according to historic data. That allows seamless changes of network configurations from end-users' perspective. Next, physical and virtual network elements are actually configured via the exploitation of reservation protocols, Openflow proactive rules setting and proactive VNFs placement.
5) *Performance verification* – The cognitive hypervisor checks that its decisions/actions have achieved the end-to-end QoS for all end users. This phase is crucial to 'close' the autonomic loop, since the autonomic entity managing the network needs to learn either from previous actions or 'errors'.

All the above phases provide new knowledge to the learning algorithm. The learning process and its efficiency strictly depend on the specific algorithm chosen during AMVNO design and implementation. However, the suggestion of specific algorithms and techniques for hypervisor's phases and learning is clearly out of the scope of this article. For some hints in this direction, the reader might refer to [1].

IV. Design and Performance Metrics

Future 5G and B5G networks are focusing on the support of three main types of communication services: Extreme Mobile Broadband (xMBB), ultra-reliable Machine-Type Communications (uMTCs) – also called ultra-reliable low-latency communications (URLLCs) – and massive Machine-Type Communications (mMTCs). All the three categories (and especially in uMTCs/URLLCs) will need to support services with very strict requirements in terms of bandwidth, latency, reliability and availability.

By now, 5G and B5G key performance indicators (KPIs) are based on the following requirements:

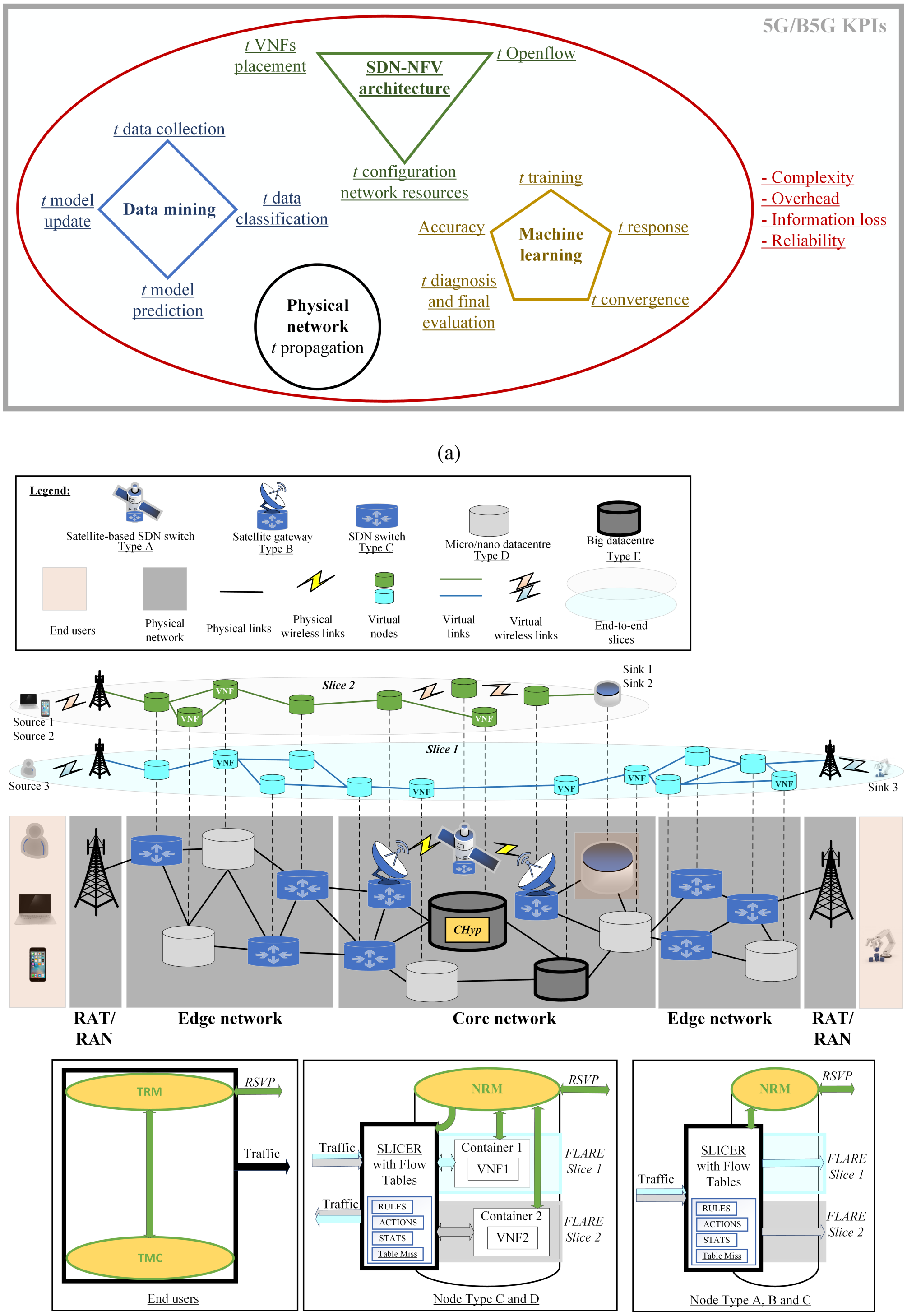
- 10 times higher data rates to individual end users;
- 1 – 10ms round-trip time;
- higher bandwidth per unit of area and enormous number of connected devices;
- perceived network availability of 99.999%;
- reduced time to set up a service from local application to network individual service components.

The main paradigm, that have been identified by research community to be capable to guarantee very high adaptability, proactive response to different situations and acceptable costs, is autonomics (see *"5G-PPP Working Group on Network Management and QoS"*).

In such scenario, 5G and B5G KPIs represent the high-level goals the autonomic system have to satisfy. Nevertheless, there are several aspects to take into account during design and implementation of such an AMVNO. Such aspects can be associated to four functionalities of the network infrastructure (see Figure 6(a)): SDN-NFV architecture, data mining, machine learning and physical network.

An AMVNO that wants to act proactively should take into consideration relevant parameters related to its background *SDN-NFV architecture*. The time to configure network resources via controllers (e.g. TCs, ICs and EMs) after decisions is fundamental to guarantee effective and 'seamless' response to dynamic network changes. Specifically, both the time to slice/to manage resources (via Openflow) and to place/to activate VNFs are of paramount importance. In particular, Openflow proactive rule placement should be employed, which takes advantage on predictive models. Such elements should be designed by considering constraints in terms of complexity, communication overhead, information loss and reliability. In fact, the techniques would need to strike a balance between the level of complexity (not to impact on network and operating costs) and overhead (not to impact on performance). Moreover, autonomic solutions have to guarantee negligible loss rates and reliability of Openflow and VNFs deployment to avoid network configuration issues. In this framework, a valuable work, which studies the reliability of VNFs considering reliability at multiple layers, can be found in [10].

As previously discussed in Section III, a fundamental phase of autonomic network management is *data mining*. The time to collect and to classify data significantly affects AMVNO delay to address network unexpected states. Furthermore, such data represent the key to improve models/patterns accuracy to understand and to forecast future network behaviours. Networks are complex systems: the huge amount of current/historic stored data requires a lot of processing effort (i.e. time) to support effective predictions. Indeed, if a model/pattern requires adjustments, its updating time can affect system performance.



(a)

(b)

Fig. 6. (a) Main domains that affect AMVNO's performance and respective key design and performance metrics (b) The example shows an InP hosting an AMVNO, which has to manage two kinds of services: uMTC/URLLC (Slice 1) and xMBB (Slice 2). The former consists of a human (Source 3) controlling remotely a machine (Sink 3), while the latter is a real-time video transmission from a centralised server (Sink 1-2) to mobile users (Source 1-2). The cognitive hypervisor is placed in a big datacentre, which support processing, storage and bandwidth requirements. It can communicate with all the nodes to get/to send information via NRMs and TRMs. TMC is responsible to collect information at the end user to be then transmitted to the cognitive hypervisor via its TRM.

From this discussion, it clearly appears how data collection and processing are closely related to complexity, overhead and reliability. Huge data load needs significant storage capacity and available processing resources. Because of that, a cognitive hypervisor should be placed in a cloud datacentre. Finally, it is important to notice the role of errors-free communications for data collection and reliability of theoretical models to avoid prediction errors.

Since autonomics implies no human intervention, the performance of *machine learning* has a central role. However, some related parameters can negatively affect the performance of AMVNOs in terms of latency:

- The collection of new data, which can change predictive models, requires time for continuous training;
- the response time, which is required to make predictions after training, affects the latency to make a decision;
- the convergence time consists in how fast an algorithm agrees that a found solution for a particular problem is the optimal one at that time;
- the time to diagnose network status and to evaluate the correctness of made decision to achieve the goals.

The main way to reduce the effect of these delays is to rely completely on predictive models to make a-priori decisions. Nevertheless, that arises other significant issues, affecting the AMVNO. The accuracy and reliability of predictive models have a key role especially when end-to-end services affect people safety or can impact on the physical world. Regarding learning algorithms, the convergence reliability is also significant, which represents the susceptibility of an algorithm to be stuck at local minima and how can initial conditions affect its performance. Finally, complexity of learning algorithms should be reduced since it can have negative effects on the system in terms of efficiency and costs. Some examples of candidate learning schemes for cognitive/autonomic networking were proposed by the authors in [11], [12], also demonstrating the flexibility of such schemes.

Finally, the physical network's parameters also affect the performance of AMVNO. Especially, the propagation time has a main role when networks like 5G and B5G involve satellites and satellite links. The set up of satellite-based SDN switches and satellite gateways can raise latency and influence reliability. In order to mitigate that, efficient algorithms should be designed to avoid performance degradation: an recent interesting work in that context can be found in [13].

V. Exemplar Use Case of AMVNO

Figure 6(b) depicts an example of 5G/B5G physical network of an InP hosting an AMVNO.

The combination of service reservation procedure, traffic engineering and data mining allows the cognitive hypervisor (inside a big datacentre) to associate mobile customers to a virtual slice with specific VNFs, which can guarantee the requirements for their end-to-end applications. The service reservation protocol is the means, which allows the control plane to reserve physical resources that are then translated into virtual network elements via Openflow commands in the flow tables. Specifically, the example suggests the possibility to use RSVP-TE with its extensions. Terminal reconfiguration managers (TRMs) and network reconfiguration managers (NRMs) are the devices' internal elements to communicate with resource reservation, autonomic control plane and SDN-NFV control plane, which exploit Openflow as their actuator in the virtual domain.

Each application session owns a data flow, which passes through a specific path. This path is translated to a dedicated end-to-end virtual network slice with specific VNFs (see Figure 6(b)). The SDN-NFV controllers (e.g. ICs, TCs and EMs) set up the flow tables and activate the VNFs belonging to the path.

In our previous real experiments in [14], we found end-to-end slicing with VNFs' activation can be efficiently realised via FLARE (a programmable node architecture) hosting containers based on Docker. Thus, the AMVNO can exploit FLARE architecture to create efficiently end-to-end slices.

Figure 6(b) shows the logic representation of a network nodes with active slices and VNFs. Physical resources at the node are separated into FLARE slices, each of which activates a container with a specific VNF. NRMs sets up the configuration according to the guidelines sent by the cognitive hypervisor. Next, the Slicer creates the virtual network topology, it receives the ingoing traffic and forwards it to the correct FLARE slice. The isolated VNF inside Docker receives the respective traffic and performs its operations, then the Slicer send processed traffic on the outgoing link. Once slices are active, the NRM updates information in the Slicer. Slicer update via NRMs can be proactive by employing proactive Openflow rules placement.

VI. Challenges and Future Directions

The realisation of an AMVNO requires stable and efficient SDN-NFV virtual infrastructures, which are still under research. However, some steps forward in the proposed direction were already been analysed in an early work [15]. The followings are the main challenges for an AMVNO grouped by technology.

First, the identification of effective cognitive algorithms for autonomic networks is still at its beginning. No significant real results are available in the literature (except [11], [12]) and standardisation activities are ongoing. Moreover, no significant discussion has been undertaken about network management without human intervention. An important aspect to consider is the amount of resources (e.g. computing and storage) and traffic load for data gathering a real-time autonomic network management requires (see

"5G-PPP Working Group on Network Management and QoS"). In case of AMVNOs managing large networks, the scalability of data mining and learning algorithm becomes important. Another aspect to consider is the training time and its relationship with the reservation procedure. That also affects the response time, which involves the efficient use of the backup list of paths. The type and amount of training data the algorithm requires to achieve high performance is also fundamental.

Next, the design of accurate and reliable predictive models for autonomic management of complex systems like 5G/B5G networks is still in its infancy.

The deployment of SDN/NFV raises the important question about centralised versus distributed control plane. In fact, an important task should be the real evaluation of which solution would optimise the performance of an AMVNO. In parallel, it would be interesting to study AMVNOs in multi-domain context.

Slicing wireless networks, base stations' resources and wireless links is still an ongoing research area. How wireless resources are modelled highly changes slicing techniques and their performance. This highlights the need to focus on RAN and Edge functions virtualisation and their real impact on AMVNOs.

The structure proposed in [14] can be an efficient approach to achieve effective isolation to prevent the deterioration on the performance of one slice because of any change in another slice (i.e. the number of end users, flows or channel conditions).

VII. Conclusions

At the best of authors' knowledge, this is the first work, which has proposed the deployment of AMVNOs for future 5G and B5G networks. This is expected to reduce costs and to increase operators' revenues. The article has provided a detailed description of AMVNOs by focusing on architecture, performance, metrics and real implementation guidelines.

This paper should be intended as a roadmap for the development of an actual AMVNO architecture since research and standardisation on that is still at its beginning and several presented issues require further investigation before real implementation will become possible.

References

[1] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A survey of machine learning techniques applied to self organizing cellular networks," *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, pp. 1–1, 2017.

[2] L. Duan, J. Huang, and B. Shou, "Cognitive mobile virtual network operator: Investment and pricing with supply uncertainty," in *2010 Proceedings IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[3] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer*, vol. 36, no. 1, pp. 41–50, Jan. 2003.

[4] B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *IEEE Communications Surveys Tutorials*, vol. 16, no. 3, pp. 1617–1634, Third 2014.

[5] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. D. Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 236–262, Firstquarter 2016.

[6] J. Matias, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an SDN-enabled NFV architecture," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 187–193, Apr. 2015.

[7] Q. Duan, N. Ansari, and M. Toy, "Software-defined network virtualization: an architectural framework for integrating SDN and NFV for service provisioning in future networks," *IEEE Network*, vol. 30, no. 5, pp. 10–16, Sep. 2016.

[8] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80–87, May 2017.

[9] M. Richart, J. Baliosian, J. Serrat, and J. L. Gorricho, "Resource slicing in virtual wireless networks: A survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 462–476, Sep. 2016.

[10] J. Liu, Z. Jiang, N. Kato, O. Akashi, and A. Takahara, "Reliability evaluation for NFV deployment of future mobile broadband networks," *IEEE Wireless Communications*, vol. 23, no. 3, pp. 90–96, Jun. 2016.

[11] C. Facchini and F. Granelli, "Towards a model for quantitative reasoning in cognitive nodes," in *2009 IEEE Globecom Workshops*, Nov. 2009, pp. 1–6.

[12] C. Facchini, F. Granelli, and N. L. S. da Fonseca, "Identifying relevant cross-layer interactions in cognitive processes," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, Dec. 2010, pp. 1–6.

[13] J. Liu, Y. Shi, L. Zhao, Y. Cao, W. Sun, and N. Kato, "Joint placement of controllers and gateways in SDN-enabled 5G-satellite integrated network," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 2, pp. 221–232, Feb. 2018.

[14] A. Nakao, P. Du, Y. Kiriha, F. Granelli, A. A. Gebremariam, T. Taleb, and M. Bagaa, "End-to-end network slicing for 5G mobile networks," *Journal of Information Processing*, vol. 25, pp. 153–163, Feb. 2017.

[15] C. Facchini, F. Granelli, and N. L. S. Fonseca, "Cognitive service-oriented infrastructures," *Journal of Internet Engineering*, vol. 4, no. 1, Dec. 2010.

Fabrizio Granelli Fabrizio Granelli is Associate Professor at the Dept. of Information Engineering and Computer Science (DISI) of the University of Trento (Italy). From 2012 to 2014, he was Italian Master School Coordinator in the framework of the European Institute of Innovation and Technology ICT Labs Consortium. He was Delegate for Education at DISI in 2015-2016 and he is currently member of the Executive Committee of the Trentino Wireless and Optical Testbed Lab. He was IEEE ComSoc Distinguished Lecturer for 2012-15 and IEEE ComSoc Director for Online Content in 2016-17. Prof. Granelli is IEEE ComSoc Director for Educational Services for 2018-19 and coordinator of the research and didactical activities on computer networks within the degree in Telecommunications Engineering. He was advisor of more than 80 B.Sc. and M.Sc. theses and 8 Ph.D. theses.

He is author or co-author of more than 200 papers published in international journals, books and conferences in networking, with particular reference to performance modelling, cross-layering, wireless networks, cognitive radios and networks, green networking and smart grid communications.

Riccardo Bassoli Riccardo Bassoli received his B.Sc. and M.Sc. degrees in Telecommunications Engineering from University of Modena and Reggio Emilia (Italy) in 2008 and 2010 respectively. Next, he received his Ph.D. degree from 5G Innovation Centre (5GIC) at University of Surrey (UK), in 2016, with his theses entitled 'Network Coding for Efficient Vertical Handovers'. He is currently postdoctoral researcher at Department of Information Engineering and Computer Science (DISI), at University of Trento (Italy). He was a Marie Curie Early Stage Researcher in the GREENET project, funded by European Commission. During that period he was working at Instituto de Telecomunicações (Portugal) and visiting researcher at Airbus Defence and Space (France). Now, he is actively involved in Dynamic Architecture based on UAVs Monitoring for border Security and Safety (DAVOSS) project, funded by NATO. His main research interests include theoretical aspects of software-defined networking, network function virtualisation, network slicing and autonomic networks.