A QoE-oriented Cognition-based Management System for 5G Slices: The SliceNet Approach

Albert Pagès, Fernando Agraz, Salvatore Spadaro, Rafael Montero
Optical Communications Group (GCO)
Universitat Politècnica de Catalunya
Barcelona, Spain

Dean Lorenz, Kenneth Nagin
IBM Research
Haifa, Israel

Rui Pedro, Guilherme Cardoso, Pedro Neves, Nuno Henriques
Altice Labs SA
Aveiro, Portugal

Xenofon Vasilakos, Nasim Ferdosian, Navid Nikaein
Mobile Communications Department
EURECOM
Sophia Antipolis, France

Marouane Mechteri, Yosra Ben Slimen
Orange Labs
Châtillon, France

Qi Wang, José M. Alcaraz Calero, Antonio Matencio Escolar, Pablo Salva, Enrique Chirivella Perez, Ricardo Marco Alaez
School of Computing, Engineering and Physical Sciences
University of the West Scotland
Paisley, United Kingdom

***Abstract*—Provisioning of network slices with appropriate Quality of Experience (QoE) guarantees is one of the key enablers for 5G networks. However, it poses several challenges in the slice management that need to be addressed to achieve an efficient end-to-end (E2E) services delivery. These challenges, among others, include the estimation of QoE Key Performance Indicators (KPIs) from monitored metrics and the corresponding reconfiguration operations (actuations) in order to support and maintain the desired quality levels. In this context, SliceNet provides a design and an implementation of a cognitive slice management framework that leverages machine learning (ML) techniques in order to proactively maintain network conditions in the required state that assures E2E QoE, as perceived by the vertical customers.**

Keywords—5G slices; network management; cognition; Quality of Experience (QoE)

I. INTRODUCTION

Future 5G networks will need to support many configurations, which leads to a multitude of system states. Under such scenario, it is no longer possible to foresee what needs to be done on every conceivable system state. As such, management and control mechanisms should be flexible enough to handle the states that have never been encountered before and for which there are no specific rules (e.g., by applying rules from similar states); thus, they must be able to generalize rules and comprehend their intention.

In addition, the sheer scale of 5G networks (up to thousands of system elements) makes it impractical (even from an economical perspective) to require human input in the control loop. Hence, such control loop has to be autonomous and automated in which the only role of humans is to prescribe the desired behavior (e.g., rules and policies). This arises from the fact that 5G networks need to support a combination of many types of workloads, stemming from a variety of use cases (UCs). These workloads vary and may even change dynamically as needed by the verticals. As a result, the derived requirements from the network may change often and significantly. Hence, in order to efficiently support them, management systems cannot only focus on the common case; they must adapt to changes and even anticipate them.

Besides the dynamicity associated to 5G networks, another important challenge related to their management is the layered architecture of their management and control solutions, involving multiple entities and roles. There are multiple information owners providing data sources, each with its own semantics. Moreover, depending on a specific role, only partial information may be available. For example, a Network Service Provider (NSP) that owns the infrastructure and implements a network sub-slice (NSS) might only provide partial network information to the Digital Service Provider (DSP) that manages the end-to-end (E2E) network slice (NS). Thus, the management and control associated to each of such roles must combine multiple information sources, interpret their meaning, and fill in the gaps (“guess”) when needed.

In addition, there is an increasing trend on shifting the focus of network services provisioning from a network perspective (Quality of Service (QoS)) to an end-user point of view (Quality of Experience (QoE)). Hence, to support service-level E2E QoE, a collection of complex tasks is required. These tasks have to translate the vertical E2E UC requirements into network Service Level Agreements (SLAs) and concrete Key Performance Indicators (KPIs), estimate and predict QoE KPIs from network-level QoS metrics, and optimize the network resources in order to support the needs of NSes.

The above-mentioned challenges can be summarized in the following list that modern management systems for 5G networks need to address and derive appropriate solutions in order to overcome them, namely:

- Flexibility: support for heterogeneous configurations and system states.
- Scalability: support for automatic management in presence of thousands of system elements.
- Dynamicity: support for varying workflows and requirements, stemming from multiple UCs.
- Abstraction: support for management and control operations as well as information abstraction across multiple stratums and roles.
- QoE and QoS: support for QoE-oriented provisioning of NSes, determining the appropriate network-level QoS.

Cognitive network management addresses these challenges by utilizing machine learning (ML) techniques in order to understand the network behavior and to proactively steer that behavior towards its desired state [1]. Given such principle, in this paper we present the SliceNet approach for a cognition-based management of 5G network slices with QoE guarantees. SliceNet advocates a declarative rather than an imperative approach to network management, where cognition is used in order to learn the best actions that achieve the declared goals. Next section describes in details the approach followed by SliceNet in regards of the whole cognition-based management loop, elaborating on the concrete strategies adopted to efficiently solve all the stages.

II. THE SLICENET COGNITION PLANE

SliceNet is a European project focused on the provisioning of multi-domain E2E 5G NSes with QoE guarantees to meet the requirements posed by key vertical UCs stakeholders (e.g. eHealth, power distribution, smart cities) [2]. To this end, SliceNet intends to meet the challenging requirements from the management and control planes of network slicing across multiple administrative domains. In particular, the objectives are: facilitating early and smooth adoption of 5G slices for verticals to achieve their demanding UCs, and managing the QoE for slice services. SliceNet follows an architectural approach suitable for both NSP and DSP business roles allowing the creation of a modular, extensible and scalable framework, encompassing both management and control layers across multiple roles within the 5G network ecosystem. Within the definition of its management solution, SliceNet proposes the incorporation of a cognition-based plane with the focus to aid on the management of 5G NSes, with QoE guarantees on the provided NSes and services on top as the main pillar of the designed cognition plane architecture.

Given this prime goal, SliceNet's Cognition Plane (CP) embraces the MAPE-K (Monitor-Analyze-Plan-Execute governed by a Knowledge-base) loop approach for the management of 5G NSes. To this end, SliceNet is designed to support ML for the Monitoring and Analysis steps, as well as for creating new Knowledge. SliceNet QoE Monitoring [3] separates the acquisition of monitoring data from the processing phase and transforming it into NS QoE metrics. The Analysis step uses the acquired knowledge to assess the NS QoE and possible impact on corrective actions. This is done by both inferring learned cognitive models and by applying more traditional automated management methods. The Planning and Execution steps (termed Actuation) are governed through a rule-based (i.e., policies) system.

SliceNet employs a Data-Driven Network Operations methodology, known as AIOPS (Artificial Intelligence for IT Operations) [4]. Network analysis applications react to operations data (both raw and processed) and generate new metrics and signals (e.g., QoE metrics and QoE-aware insights) that in turn trigger network operation actions. With this methodology, most components interact only with the data, acting as consumers and producers. This approach minimizes direct interfaces, providing flexibility and an easier integration of cognitive tools. It also allows existing techniques to be used with little change, as the outputs of the cognitive tasks can be treated as advanced sensor metrics.

The main components of the CP and its relation to the other SliceNet planes are shown in Fig 1. This figure depicts the SliceNet logical components that have significant interaction with the CP; other components of the SliceNet logical architecture are hinted by empty boxes (the interested reader

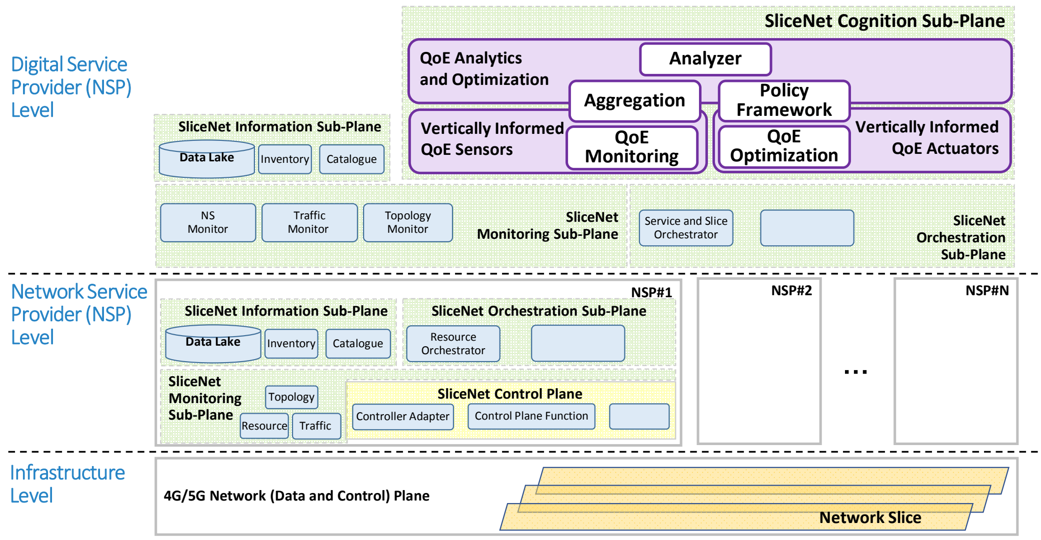

Fig.1 SliceNet's architecture and Cognition Plane overview

can refer to [5] for a complete description of SliceNet architecture).

Vertical informed sensors provide QoE monitoring capabilities and prepare aggregated data from the ML-based analytic processes. Monitoring capabilities follow a Data-Lake approach [6], having direct interfaces only with the persistent data stores. Analytics and optimization tasks process the QoE sensor data to provide cognitive insights that enable the vertical-informed actuation. Analysis is applied first to historical data to learn new models and derive new policies and optimization parameters. At run-time, the learned models are inferred in near-real-time to provide input signals for actuation. A Policy Framework (PF) governs the actuation process, applying QoE optimization tasks that combine inferred signals with measured KPIs to control vertical informed QoE actuators. Actuation may trigger new analysis or (re-)configurations for the provisioned E2E NSes.

Given the presented design and approach, the following sub-sections provide more details about the overall adopted MAPE-K loop and its main components exercised through the presented architecture.

A. *Cognitive Control Loop*

SliceNet employs a Proactive Control Scheme (PCS) for managing the lifecycle of NSes (based on the MAPE-K loop), while utilizing cognitive methods to maintain the desired SLAs. The main control loop of the PCS is portrayed in Fig. 2.

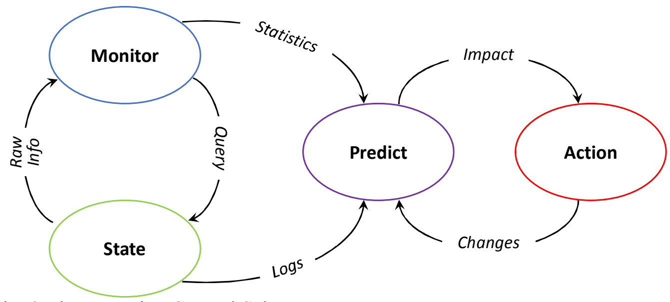

Fig. 2 The Proactive Control Scheme

Monitor and *State* components generate the data required for analysis. The *Monitor* component is continuously querying the current *State* to extract raw data about the NSes. In addition to the raw metrics, *Monitor* component generates calculated KPIs by processing the raw data as network statistics and by applying aggregations at different levels.

Analytics and ML are encapsulated in the *Predict* component. It further enriches data, adding insights, predictions, and impact analysis, to allow proactive management with better system diagnostics and prognosis. The calculation infers already learned ML models, combining real-time monitored data with log data and historical data. The enrichment information is added to the monitored data as an extra source for the *Action* component.

The *Action* component has a dual role; (1) an internal ML control loop; and (2) a network control loop for taking network actions. The ML control loop can imply actions such as (re-)validating a ML model in use, changing monitoring granularity, (re-) starting learning, etc. Network control loop actions, on the other hand, include examples such as a changing core network (CN) resources, migrate an NSS from one NSP to another and so forth. Finally, the changes caused by actions are fed back to *Predict* component alongside raw data and processed statistics in order to maintain ML models, thus closing the loop and allowing for consecutive iterations of the monitor-predict-action cycle.

B. *Knowledge and Monitoring*

SliceNet employs a Knowledge-Centric, Data-Driven approach to network operations (see Fig. 3). A logical Data-Lake data store acts as the Knowledge-base (KB) of the MAPE-K loop. All data sources are logically merged into one data store and analysis outcomes are shared through the same data store. This paradigm is used both at the NSP level to manage NSSes deployed at its underlying infrastructures, to offer a Network Slice as a Service (NSaaS) towards the DSP level, which then constructs and manages E2E NSes and offers an NSaaS to its vertical customers.

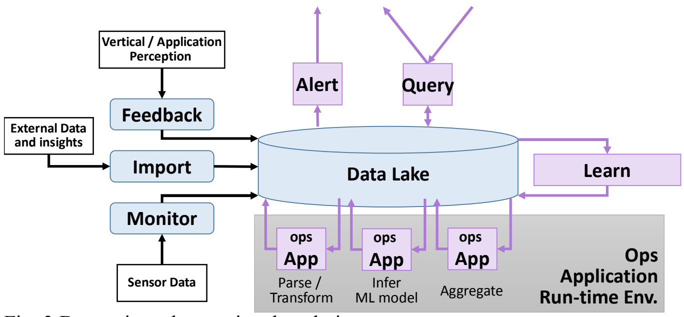

Fig. 3 Data-oriented operational analytics

Multiple data sources are logically merged to provide all the required information for QoE management. Control plane and data plane sensor outputs are collected and persisted to support monitoring operations through parsing, transformation, and aggregation. However, this data is also used for ML model training and for extracting QoS metrics. Feedback from the vertical is combined to allow the data processing application to assume the role of QoE sensors, learning and estimating the vertical perspective. This allows for QoE NS management under practical limitations, where some information must be curated to hide sensitive data or anonymized.

Data-operations applications may be deployed for each NS/NSS to filter relevant data, apply security, add context, aggregate slice metrics, etc. This addresses several of the design challenges related to the monitoring framework of 5G slices. Moreover, flexible QoE sensors may be employed; from simple aggregation and transformation tasks to inference of elaborate ML models.

C. *Analysis*

Having in mind the degree of flexibility that NSes brings to the network management domain, it can be foreseen that dealing with such an environment brings a new set of challenges while trying to apply ML techniques. Due to the presence of these challenges, the need for having an automated ML pipeline arises, not only to deal with network dynamicity but also with the problem scopes that come from different business roles (i.e. DSP and NSP). Thus, SliceNet adopts a ML pipeline to automatize the inference, deployment, execution and refinement of models that aid on the management of NSes on the underlying 5G shared network infrastructure. Fig. 4 depicts a schematic of the proposed pipeline.

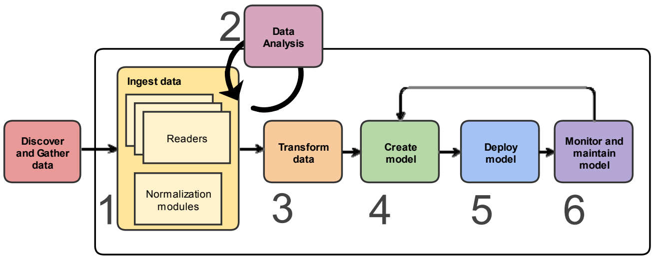

Fig. 4 SliceNet's ML pipeline architecture

As a starting point, and external to the pipeline, there is a data discovery and gathering phase; this is where input for ML occurs. Logically, this step represents a data source from the pipeline point of view. Internally, it is divided into six different functional areas, covering all phases from data collection to the ML models' lifecycle:

1. **Ingest data**: this module enables the pipeline reading data and its responsibility is divided into two components:

a. *Readers*: data input can be multiple sources containing observations or streaming data. Each reader abstracts the medium source of the observations and their nuances.

b. *Normalization modules*: data normalization is the process of combining, merging and cleaning, according to the knowledge gathered from its analysis. It includes removing duplicate observations, invalid and/or badly formed data.

2. **Data analysis**: the initial analysis serves the purpose of gaining data insights and further problem contextualization. This module runs statistical queries (e.g. counting, averaging), to check if the dataset is balanced, incomplete or how to focus its modelling.

3. **Transform data**: data transformation depends on data analysis and problem objectives. This module transforms the data into ML-ready. This is where features are extracted and their normalization (e.g. ordinal, one-hot encoding) happens.

4. **Create model**: ML algorithms, which can cover the classification, prediction or clustering ML areas, are applied in this phase. This is where models are effectively trained, optimized and their testing strategies are put in place.

5. **Deploy model**: during the training/testing phase, if a model shows significant fitness metrics values it can then be deployed into production and start being used to predict, classify or cluster data in the real-time problem domain.

6. **Monitor and maintain model**: deployed models can lose their effectiveness over time, especially when the data domain is too volatile and dynamic. This means that certain models may be unfit for usage since they no longer properly represent the real world. When models show fitness metrics that are below the configured acceptable values, they are archived, and a re-training task is scheduled to update them.

The CP architecture uses data processing applications not only to process data for training, but also to infer the learned models at slice run-time. This approach provides a flexible run-time environment that can support complex cognitive models as well as simple rule-based optimizations. It also has the advantage that the processing resources required to build the analytic data pipeline can be attributed to the slice(s) that consume the resulting information. The orchestration and life-cycle management of these CP applications has similarities with the Network Function Virtualization (NFV) Management and Orchestration (MANO) architecture [7]. As such, the analytic models are just functions that process data events and are thus compatible with a Function-as-a-Service (FaaS) execution environment. Finally, more traditional control mechanisms (such as threshold-based triggers and rule-based alarms) and management utilities can also be implemented as data-processing applications.

D. Planning and Execution

The complexity and dynamicity of NSes require autonomous loops for enforcing concrete actions on the underlying infrastructure to ensure an optimal system performance. Moreover, given the more user-oriented perspective of 5G networks, both planning and execution phases must be built around the concept of quality optimization at all levels. SliceNet defines an actuation framework with the core goal of maintaining and optimizing the perceived QoE by vertical customers. To this goal, the actuation framework focuses on determining the required changes to E2E NSes that support the verticals' services, while taking charge of enforcing such changes. The actuation framework is designed with two main components in mind, namely:

1. **Policy Framework** (PF): is a rule-based policy engine, in which rules define what actions are executed in response to system and NS events.

2. **QoE Optimizer:** responsible for all (re-)configurations necessary to maintain the QoE of a specific E2E NS. Thus, given the rules specified by the PF, and gathered monitoring data, the QoE Optimizer triggers the necessary actions to carry out the desired actuations.

The PF and QoE Optimizer get their input from both analytics and external monitoring data to determine when and how actuations should be carried out. Then, the actual actuation is triggered by timely collaborations across functions at different layers. In a nutshell, the rules defined by the PF are translated into operations that the QoE Optimizer can trigger/execute. These trigger/executions are abstractions of the operations exposed by execution points (mainly, control and orchestration layers). Traditional Operations Support Systems (OSS) are evolving towards more flexible, agile and service-oriented management platforms. This can be achieved through policies, which can be seen as high-level directives that convey what the software components should do under certain conditions. The PF designed in SliceNet enables the project system architecture with policy-driven capabilities. Fig. 5 depicts the high-level view of the PF.

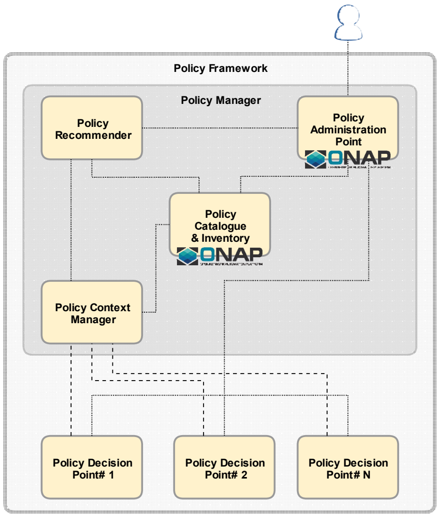

Fig. 5 SliceNet Policy Framework architecture

SliceNet's PF design and software implementation is partially aligned with ONAP Policy Subsystem [8]. In summary, policies are administrated through a central Policy Administration Point (PAP) component, which enables the on-boarding of policies by a system administrator, their parametrization/tuning and their dissemination towards Policy Decision Points (PDPs). It is in these PDPs where deployed policies are executed during the run-time of NSes. As such, policies dictate the system behavior as indicated in the policy parameters - for example, Event-Condition-Action (ECA) policies that implement closed management/control loops in the architecture (e.g. imminent faults mitigation).

Acting as a PDP, SliceNet's QoE Optimizer component is responsible for triggering the desired actions (Execution from the MAPE-K loop) within the actuation framework. The actions are meant to maintain the quality of a particular E2E NS deployed on the underlying infrastructure, which may encompass several NSPs/segments/domains. As such, the QoE Optimizer is designed as a module that will be instantiated per E2E NS. A QoE Optimizer instance will have a specific actuation scope tailored to its NS, at the DSP level, since it is necessary to gain visibility of all elements/NSSes that intervene and may affect the quality of the delivered NS.

To achieve this goal, the QoE Optimizer (Fig. 6) follows a simple approach. On the one hand, it listens to monitoring data relevant to the quality of the NS under its responsibility. On the other hand, actions are directed to different parts of the SliceNet ecosystem to enable (re-)configurations at the underlying NS. These actions may be applied at concrete parts (thus, enforced to specific NSPs) within the E2E NS or affecting the totality of the slice as a whole. The QoE Optimizer behavior is rule-based, with rules being the policies disseminated by the PF. For each of the monitoring events that the concrete instance of QoE Optimizer is subscribed to, the policies define a specific condition that should be checked periodically. When a condition is violated, a corrective action is applied. This action is specified by the policy and the QoE Optimizer is responsible for coordinating the actuation workflow that fulfils the desired (re-)configuration operation.

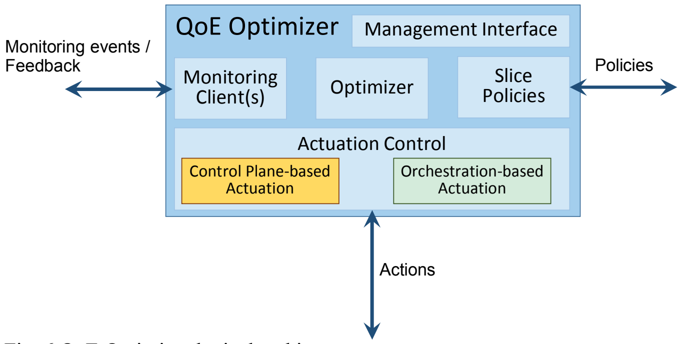

Fig. 6 QoE Optimizer logical architecture

The consequences of the actions are then captured by the monitoring system and feed again to the centralized Data-Lake for their further analysis and future actuation, closing the cognition loop.

III. CONCLUSIONS

One of the key features of future 5G networks is to provide an automated and QoE-aware management of E2E NSes as offered by DSPs entities towards vertical customers. To achieve this goal, it is essential to provide means for analysis of the underlying NSes and of their components in order to determine their quality levels and to apply (re-)configurations when needed to maintain the desired quality levels.

In this context, SliceNet's CP allows for a holistic and automated management of E2E NSes. SliceNet has designed a novel policy-ruled actuation framework, which determines when and how (re-)configurations should be made on the underlying NS/NSSes. This is enabled thanks to the data-centric and analysis approach followed by the whole CP in which all sources of monitoring are logically centralized into a single source, allowing for external elements to gather desired data, elaborate on that, and insert it again in the unified Data-Lake for later consumption by other elements of the CP.

ACKNOWLEDGMENT

The work in this paper has been supported by SliceNet EU H2020 project (H2020-ICT-2016-2/761913).

REFERENCES

[1] 5GPPP Network Management & Quality of Service Working Group, "Cognitive Network Management for 5G", white paper, March 2017.

[2] The SliceNet project, https://slicenet.eu/

[3] SliceNet Deliverable D5.2, "Modelling and Design of Vertical-Informed QoE Sensors"; SliceNet Consortium, December 2018.

[4] Andrew Lerner, "AIOps Platforms", Gartner Blog, August 2017. [online]: https:// blogs.gartner.com/andrew-lerner/2017/08/09/aiops-platforms/

[5] SliceNet Deliverable D2.2, "Overall Architecture and Interfaces Definition"; SliceNet Consortium, January 2018.

[6] N. Miloslavskaya, A. Tolstoy, "Big Data, Fast Data and Data Lake Concepts", Elsevier Procedia Computer Science, vol. 88, pp. 300-305, October 2016.

[7] ETSI, "Network Function Virtualization (VNF); Management and Orchestration", December 2012.

[8] ONAP Policy Framework [online]: https://wiki.onap.org/display/DW/The+ONAP+Policy+Framework