IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 March 2023, accepted 10 April 2023, date of publication 17 April 2023, date of current version 21 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3267548

RESEARCH ARTICLE

Digital Twins for 5G Networks: A Modeling and Deployment Methodology

MARIO SANZ RODRIGO[1], **DIEGO RIVERA**[1], **JOSÉ IGNACIO MORENO**[1], **(Senior Member, IEEE), MANUEL ÁLVAREZ-CAMPANA**[1], **AND DIEGO R. LÓPEZ**[2]
[1]Departamento de Ingeniería de Sistemas Telemáticos, ETSI de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]Telefónica I+D, 28010 Madrid, Spain

Corresponding author: Mario Sanz Rodrigo (mario.sanz@upm.es)

This work was partially funded by Spanish Ministerio de Asuntos Económicos y Transformación Digital through UNICO-5G I+D program (NextGenerationEU), within the B5GEMINI-INFRA project (Beyond 5G Enhanced Management through dIgital twiNs based on artificial Intelligence), under Grant TSI-063000-2021-81.

ABSTRACT A Mobile Networks Digital Twin (MNDT) is a virtual replica of a mobile communication network that accurately models the devices, the communication links, the operating environment, and the applications that run on the physical network. By replicating different environments in a laboratory and running multiple scenarios, Digital Twins provide a cost-effective way to evaluate performance, predict the effects of network changes, optimize network management, and make appropriate decisions. This paper presents a methodology for automatically creating and using Network Digital Twins, along with a proposed architecture for performing this implementation. This work is framed in the context of the B5GEMINI project, whose goal is to develop an MNDT applied to a 5G core environment and its evolution towards 6G, being able to apply it to use cases in advanced scenarios such as cybersecurity or Industry 4.0. The proposed methodology covers the entire lifecycle of the MNDT, from the initial phases of data acquisition and modeling to the phases of use and bidirectional connection between physical and digital elements.

INDEX TERMS 5G core, architecture, digital twin (DT), mobile networks digital twin (MNDT), design methodology, management and orchestration (MANO), network function virtualization (NFV).

I. INTRODUCTION

In recent years, Digital Twin technology has emerged as a new paradigm for the design, deployment, and operation of next-generation networks. This new technology is based on accurately modeling the physical network to create its digital counterpart, and establishing a bi-directional link between the two twins so that they can evolve synchronously throughout their lifecycle.

This new paradigm is being applied in many fields such as Industry 4.0, cybersecurity, healthcare, aerospace, intelligent systems, 5G and 6G networks, etc., [1], [2], [3], [4], and [5].

This paper addresses the concept of Digital Twin applied to mobile networks, called Mobile Network Digital Twins (MNDT), as a technology that enables next-generation networks and take advantage of their full potential by integrating the latest advances in 5G technology, software-defined networks (SDN), or virtualized network functions (VNF). This integration would create a complete technological ecosystem that offers major benefits in areas such as cybersecurity, preventive network maintenance, quality of service control, network optimization, and self-configuration or action planning network considering the expected behavior of network users.

Despite the numerous advantages offered by this new paradigm, one of its main challenges is the heterogeneity of the equipment that compose the new operator networks, leading to the definition of new methodologies focused on the collection of a specific data nature, for the creation of a semantic representation of the network as well as the characterization of its behavior, for the subsequent modeling of the MNDT. These two issues are critical because the success of the Digital Twin technology is highly dependent on the precision in modeling the physical element, which in turn

The associate editor coordinating the review of this manuscript and approving it for publication was Rentao Gu.

requires extensive data collection for further processing, linking, and knowledge derivation. In addition to the problem related to data acquisition and Digital Twin modeling, there is another, even greater challenge related to the communications between the Digital Twin and its physical counterpart (which is, in fact, one of the keys to the new paradigm that enables bidirectional interaction between entities). Due to the sensitivity of the data exchanged between the two twins in the MNDT ecosystem, this type of connection requires the use of security mechanisms that comply with the confidentiality, integrity and authenticity (CIA) triad of data. Although not all communications within the MNDT need to proceed in this manner, it is necessary to be aware of the need for real-time communications in certain data exchanges, such as prior to a network preemptive action or related with quality of service configuration.

This work focuses on the study of different technologies that enabled the creation of next-generation networks using Network Function Virtualization (NFV), 5G, and Digital Twin technologies for their applicability in operator networks. Based on this study, we propose the definition of a methodology for the automatic creation of MNDT, which includes the phases of data acquisition, modeling, use, deployment, and interconnection between the twins to contribute to the state of the art by standardizing the modeling and use of digital twins applied to the specific use case of 5G mobile networks.

This article is organized as follows. First, a theoretical overview of the technologies that can be used for MNDT implementation is presented in Section II. Section III analyzes the main works that have chosen to apply this technology based on Digital Twins. Sections IV and V present both the methodology and the proposed architecture. Finally, sections VI draw the conclusions from this work.

II. BACKGROUND

This section presents the necessary technological background for the proposed methodology implementation, which is applied to the specific case of 5G mobile networks to model and deploy them as Digital Twins.

Four fundamental pillars are identified, the first one based on the Internet Engineering Task Force (IETF) draft about Digital Twin Networks [6], the second one based on the Stand-Alone implementations of 5G technology, the third one based on the NFV architecture proposed by European Telecommunications Standards Institute (ETSI), and finally on the adaptation of modeled elements to NFV templates.

A. NETWORK DIGITAL TWINS

The concept of the Digital Twin has been attracting interest from both academia and industry for years. We are currently at a point where technology is enabling the implementation and advancement of this type of technology, which promises great benefits in areas as diverse as real-time monitoring and optimization, risk control, cost reduction, maintenance improvement, and more.

The term Digital Twin is not new in the literature. in 2002, it was mentioned by Michael Grieves in his presentation on Product Lifecycle Management (PLM) at [7]. Later, Framing et al. proposed "an agent-based architecture in which each product is associated with a corresponding virtual counterpart or agent" [8]. These types of approaches, dating back two decades, are very similar to the current concept of Digital Twin technology, which, in short, consists of a digital image faithfully its physical counterpart.

Currently, the IETF defines the concept of a Digital Twin through the current draft "Digital Twin Network: Concepts and Reference Architecture" [6], as a real-time representation of an entity or physical element in a digital environment. This digital entity has several properties, including real-time interaction, iterative operation, and process optimization, addresses the entire lifecycle, and builds a comprehensive network based on data.

The focus of this design is not only on the general paradigm of Digital Twins, but it also specifies a special case, namely Network Digital Twins. These aim to fully model the network elements as well as the topology to obtain a dynamic emulation of the network. This approach is very different from traditional emulations or simulations that are not connected to the network they emulate or simulate. This approach, based on DT, has the unique property of establishing a real-time interactive and bidirectional communication between physical and virtual parts.

Although there is no formal definition of the Network Digital Twin concept, the IETF draft [8] defines it as a virtual representation of the physical network to be used for network analysis, diagnosis, emulation, and control. This representation includes four key elements: Data, Mapping, Models, and Interfaces.

1) **Data**. A Network Digital Twin should maintain historical data and/or real-time data (configuration data, operational state data, topology data, trace data, metric data, process data, etc.) about its real twin (i.e., the physical network) that is needed by the models to represent and understand the real twin's states and behaviors.
2) **Models**. Techniques that involve collecting data from one or more sources in the real-world twin and developing a comprehensive representation of the data (e.g., system, entity, process) using specific models. These models are used as an emulation and diagnostic foundation to represent the physical network's dynamics and elements of operation and generate data for decision-making.
3) **Interface**. Standardized interfaces can ensure the interoperability of the Network Digital Twin. There are two main types of interfaces:
   - The interface between the Network Digital Twin platform and the physical network infrastructure.
   - The interface between Network Digital Twin platform and applications.
4) **Mapping**. Used to identify the Digital Twin and the underlying entities and to establish a real-time

interactive relationship between the physical network and the twin network or between two twin networks. The mapping can be:

- One to one (pairing, vertical): Synchronize between a physical network and its virtual twin network with continuous flows.
- One to many (coupling, horizontal): Synchronize among virtual twin networks with occasional data exchange.

The NDT, built on the four fundamental technological elements, can analyze, diagnose, emulate, and control the physical network throughout its lifecycle using optimization algorithms, management methods and expert knowledge.

In relation to these four elements, the methodology Proposed in this article is adapted in the following way:

- Data: IETF pays special attention to the data collected about the physical network, since it is necessary for modeling the Network Digital Twin as well as for modeling the communications. In the proposed methodology, agents that can be integrated into the target network and are responsible for collecting this information have been designed.
- Models: due to the nature of the target networks to be modeled, data-based models are chosen to model each one of the physical elements, with the goal of creating a VNF that is configured and behaves in the same way as its physical counterpart. These models are based on deriving knowledge from the data collected in the previous acquisition phase.
- Interface: the IETF draft speaks in general terms about the need for interfaces to establish connections across the MNDT ecosystem. Specifically, the interconnection point is addressed by our proposed methodology in its interconnection phase (see section IV). Our methodology proposes using gateways at the networks' edge (both in physical and digital twin networks). These gateways are defined to be able to establish peer-to-peer relationships between physical elements and their digital counterparts using messaging protocols. Moreover, the use of MANO platforms allows the MNDT network to be connected to the rest of the applications or modules compatible with this type of technology (e.g., life cycle management modules, ML-based data processing modules, AI-based decision support modules, etc.).
- Mapping: both the IETF draft and various surveys analyzed in the context of the digital twin paradigm [1], [9] emphasize the importance of mapping between physical and digital entities. The work presented in this paper addresses this open challenge by proposing the use of gateways with messaging protocols such as Message Queuing Telemetry Transport (MQTT) to perform the mapping between the different entities. In addition to mapping, this work proposes the use of MQTT-based PQC (post-quantum cryptography) to ensure the security and robustness of the communication between the twins.

All these issues are discussed in more detail when explaining each phase of the proposed methodology in Section IV.

B. 5G TECHNOLOGY

The 5G technology, developed by 3GPP, started its specification in versions 15 and 16 and hopes to finalize it in versions 17 and 18 [9], [10], to meet the expectations of the ITU-T defined IMT-2020 evolution [11]. 5G networks are envisioned as a mobile access network technology with the intent to integrate multiple ‘‘vertical’’ environments: Industry (Industry 4.0), Vehicles (V2X), Health (eHealth), Energy (Smart Grid), etc. These environments are characterized by having very different requirements depending on the scenario (latency, scalability, availability, and reliability) compared to current networks with a ‘‘one-size-fits-all’’ architecture. To meet these requirements, 5G technology introduces groundbreaking concepts such as SDN [12], NFV [13], or Network Slicing [14] compared to previous versions.

Due to the Open-Source implementations of standalone 5G core networks, it is possible to use technology based on digital twins applied to these networks. The MNDT ecosystem would cover, in its modeling phase, heterogeneous and dynamic elements such as those that make up the aggregation network or the access network, and static elements such as the 5G core, implemented through network functions (NFs).

In a service-based architecture, the services that the 5G core need to provide in a mobile network are executed by enabling interworking between a set of services. These services are also called NFs and each of them has a different role in the core [15]. The main NFs of the 5G core are explained below:

1) **UPF**. The user Plane Function is the network function in charge of the user plane. This means the packet routing and forwarding, management of QoS, and statistics of the session.
2) **SMF**. The Session Management Function is mainly in charge of the session management, IP addressing allocation, and pricing.
3) **AMF**. The authentication Server Function is the network function in charge of the signaling plane such as the management of mobility, access, and authentication.
4) **UDM**. The Unified Data Management UDM is the NF in charge of Subscription management. Facilitates User identification and Authorization based on user data.
5) **AUSF**. The Authentication Server Function is responsible for the authentication process of 3GPP or non-3GPP.
6) **PCF**. The Policy Control Function is responsible for the unified policy management and provides policies at the user level and QoS policies for flows.
7) **NRF**. The Network Repository Function oversees the other NF discovering services that other NF can offer, it acts as a ‘‘service repository’’.

8) **NSSF**. The Network Slice Selection Function is in charge of the selecting the Network Slice of the User End based on the data obtained during the User End attachment.

There are many Open-Source projects and implementations related to the deployment of 5G technology, covering the different elements of a 5G network as well as different implementations of each service. Below are different Open-Source projects in relation to 5G technology, which cover the Radio Access Network, core network implementations, and platforms or frameworks designed for the use of 5G technology.

TABLE 1. Open-source projects about RAN 5G.

Project	License	Language	Docker Deployment
free5GRAN [16]	Apache 2.0	C++/ C	No
OAI-RAN [17]	OAI Public License v1.1	C/ Python/ C++ / Shell	No
srsRAN [18]	AGPL-3.0 License	C++ / C	No
UERASIM [19]	GPL-3.0	C ++	Yes

TABLE 2. Open-source projects about 5G core.

Project	License	Language	Docker Deployment
free5GC [20]	Apache 2.0	Go / Shell	Yes
Internship-5GCN [21]	MIT License	Java	No
OIA-CN [22]	OAI Public License v1.1	C++ / C	Yes
Open5GS [23]	AGPL-3.0	C	Yes

TABLE 3. NFV implementations support in 5G core projects.

Function	free5GC	Internship-5GCN	OAI-CN	Open5GS
AMF	Yes	Yes	Yes	Yes
UPF	Yes	-	Yes	Yes
SMF	Yes	Yes	Yes	Yes
AUSF	Yes	-	Yes	Yes
NRF	Yes	Yes	Yes	Yes
PCF/PCRF	Yes	-	-	Yes
UDR	Yes	-	Yes	Yes
UDM	Yes	Yes	Yes	Yes
N3IWF	Yes	-	-	-
NSSF	Yes	-	Yes	-

TABLE 4. Platforms with a 5G core implementation.

Project	License	Language	Docker Deployment
docker_open5gs with IMS [24]	BSD 2 Clause	Shell / Python	Yes
free5gmano [25]	Apache 2.0	Python / Go / Shell	Yes
towards5GS-helm [26]	Apache 2.0	Smarty	Yes / Kubernetes

C. NETWORK FUNCTION VIRTUALIZATION

NFV technology can be described as a network architecture model aimed at virtualizing network services. It enables the replacement of specialized hardware devices such as routers, switches, or firewalls with software-based network functions running as virtual machines on generic servers. This means that all functions executed by network nodes must be defined in a virtual model, better known as VNFs.

Due to the characteristics of this type of technology, it is the ideal candidate to enable the deployment of network elements modeled as digital twins.

The standardization of NFV is led by ETSI (European Telecommunications Standards Institute) [27]. ESTI proposes a reference architecture that defines functional blocks and interfaces for the management and orchestration of virtualized network services.

The high-level view of ETSI NFV architecture is composed of three different functional blocks: Network Function Virtualization Infrastructure (NFVI), block, VNFs block, and Management Block.

1) **Network Functions Virtualization Infrastructure (NFVI).** It forms the general basis of the architecture. It contains the hardware to host the virtual machines, the software that enables virtualization, and the virtualized resources themselves. The VNF implementation depends on the NFVI block, which provides hardware resources, the virtualization layer, and software resources to implement a particular VNF. The virtualization layer decouples the software resources from the underlying hardware and provides isolation from other VNFs while acting as an interface to the hardware resources [28].
2) **Virtualized Network Function (VNF).** This is a virtual implementation of a network function using the software. It uses the virtual machines offered by the NFVI block.
3) **Management and Orchestration (MANO).** It provides the framework for managing NFV infrastructure and virtualized network services. It includes orchestration and lifecycle management of resources interacting with both NFVI and VNFs. It is responsible for controlling all entities within the NFV architecture. The functional blocks within a MANO architecture are composed of the VNF Manager (VNFM), the NFV Orchestrator (NFVO), and the Virtual Infrastructure Manager (VIM).

D. TEMPLATES TOOLS FOR NFV SYSTEMS

Two key concepts are addressed in the context of ETSI's proposed NFV architecture. The first is the VNFD, a virtual network function descriptor responsible for describing the instantiation parameters and operational behavior of VNFs. The second is the NSD, Network Service Descriptor, which is responsible for describing the collection of configuration documents and specifying what the network service is composed of with respect to VNFs. In the case of VNFDs, these consist of Virtual Deployment Units (VDUs), which correspond to the virtual machines deployed in the NFV architecture via the VIM. The following is a brief overview of the technologies and data models most commonly used to create these types of templates, such as those supported by OSM (open source MANO) [29], [30].

1) **OASIS TOSCA** [31]. Is a model-driven language for creating templates consisting of VNF topologies, lifecycle events, and infrastructure requirements. [32] and [33].
2) **Jinja2** [34]. It is a template manager commonly used to create VNFD and NSD descriptors in YAML format.
3) **YANG** [35], [36], [37]. The YANG data modeling language is defined in IETF RFC 7950 [38]. The YANG language is used to model configuration and state data. IETF RFC 7950 states that the data model represented by YANG is meant to be manipulated by the Network Configuration Protocol (NETCONF).

Based on the analyzed background, we have concluded that there is a series of open challenges which should be covered by the different phases of the proposed methodology:

- The need for Real-time data collection mechanisms (topology, configurations, telemetry, communications, states, etc.).
- The need for collected data pre-processing, in order to perform data-based modeling capable of reflecting both the physical characteristics of the network and the behavior of its elements in response to specific data streams.
- The need for a framework for deploying and managing the 5G network twins lifecycle based on NFV architectures and aligning the models representing the physical entities with the specific data model supported by the MANO architectures.
- The development and implementation of interfaces for the secure connection between physical and digital elements.
- The need for standardization following the main work done by ETSI in the field of digital network twinning.

III. RELATED WORK

Over the past decade, the research community has become increasingly interested in technologies based on Digital Twins. As stated in an extensive work by the authors in [39], this technology is based on a digital representation of physical entities and their connection and interaction between both worlds. Currently, there are different approaches to the use of Digital Twin technology depending on the application domain, especially in the stages of twin modeling.

Although there is much literature on the general concept of Digital Twin, another approach is emerging that encompasses Digital Twin technology, giving rise to the concept of Network Digital Twins (NDTs).

The authors in [1] reinforce the key technologies associated with this type of network in areas such as communications, physical computing, DT modeling, and cloud and edge computing. Like the authors in [39], they identify several application areas among which the new generation of 5G and 6G networks stand out.

Furthermore, in [40], the authors conduct a broad study that addresses the design, modeling, and implementation of digital twins. This work covers the design of the digital twin based on functional requirements such as optimization, security, monitoring, or training, the modeling of the digital twin, or its development and deployment. Although data-based modeling is mentioned within a broad classification of models that can be applied to our work, the functional requirement associated with our proposed use case is not explicitly addressed. We are focusing on the topological representation of 5G networks and modeling their behavior, for which we intend to use models based on the interrelation of data (physical, logical, and even new inferred data obtained from ontology-based technologies). This is an open challenge that is addressed in this paper, taking into account that the modeling, data collection, digital twin deployment, etc. for networks do not present the same requirements as, for instance, industrial processes. In this work, a complete framework is presented, linked to a methodology that can be used as a basis for future validation tasks of the digital twin models generated as elements of an NFV architecture. After an analysis of the technologies required to create the digital twin, it can be concluded that current frameworks such as Eclipse Ditto [41], FIWARE [42] or the framework proposed by Albo [43] are not 100% compatible with the specific case of digital twins of mobile networks.

Moreover, in [44], the authors relate the concept of NDT to the specific application in operator networks. In this work, the authors highlight the advantages of this technology in areas such as fault location, network planning, network anomaly detection, and education and training. They propose a reference architecture for NDT that consists of three elements. First and foremost is the Network Digital Twin Model, in which a Digital Twin is created based on Deep Learning that uses inputs such as traffic, topology, routing, programming policies, etc., and provides outputs such as utilization, delay, packet loss, etc., that can be replicated in a production network based on quality-of-service contracts. They emphasize the importance of using telemetry or monitoring data, which is important in the context of Digital Twins, as the two contexts can and should cross-fertilize. Finally, they focus on the importance of data capture, both in terms of storage and the need for a standard format for capturing heterogeneous data.

Following the approach to data collection initiated by reference [44], the authors of [45] present a sketching algorithm for collecting data over the physical network. The collected data is processed by an enrichment algorithm based on the semantic representation language OWL, proposed by the World Wide Web Consortium. They also describe a proposed NDT architecture similar to the one proposed in the IETF draft [3], which provides four layers (physical network layer, Digital Twin layer, network application layer, and data lake layer).

As mentioned at the beginning of this section, one of the application areas of DT technology is 5G networks. The authors in [2] note the importance of DT technology in conjunction with 5G and artificial intelligence. This work focuses on the use of this technology to provide easy and cost-effective access to 5G technology, enabling flexible and repeatable developments over time, allowing proactive modeling of traffic, and helping to accelerate research and development time to market for new services based on next-generation networks. In addition, in the area of 5G networks, the authors in [46] address the problem of network slicing, which would allow optimization of network performance thanks to the DT technology. However, due to the interconnectedness inherent in DT technology, the authors propose DT models based on graphical neural networks (GNN) that would allow the discovery of the relationships and dependencies between network slicing, resource utilization, and the physical infrastructure associated with the physical twin.

Just as the DT technology is sparking interest in 5G, it is also sparking interest in emerging works on 6G networks. The authors in [47] propose the use of Digital Twin technology at the edge of the 6G network, achieving a combination of Multi-access Edge Computing (MEC) and DT, which would enable real-time monitoring of the entire network. MEC combined with 6G allows data to be fed directly into a decision-making module capable of performing actions on the physical network in an automated manner to optimize management.

Despite the above work, which mainly focused on NDT architectural proposals or very specific use cases, there are two papers ([48], [49]) in which the authors address an interesting concept not yet fully developed in related studies, namely a methodology for modeling and implementing Digital Twins. It should be noted that the phases of data acquisition and subsequent modeling of the Digital Twin are completely different depending on the physical element being modeled. For this reason, the presented methods are perfectly suitable for industrial processes, but if the nature of the data changes, it would be difficult to use them for modeling Digital Twins of Mobile Networks.

Finally, after reviewing the previously mentioned works and referring to two key points of this kind of technology, bidirectional communication and integration in the DT ecosystem of networks or elements in production, special attention is given to the work presented by the authors in [50], where they address security requirements such as synchronization security, communication latency, access control, network isolation or resilience. Moreover, they propose an architecture for the security application that considers both the security of the two twins and the security of the communication between the twins, which is focused through synchronization gateways.

In addition, in [40], the authors point out the open challenges related to the secure design of digital twins, focusing on the reliability of data exchanged between twins. In the specific case of modeling digital twins for 5G communication networks, this aspect can be covered by applying current hardening and security mechanisms that apply to Information and Communication Technology (ICT) environments. However, due to the critical importance of intercommunication between twins, we propose the use of post-quantum cryptography (PQC) [51], [52] on transport layer security (TLS) in the methodology phase for the twin link to try to make the communication between physical and digital elements more robust.

IV. METHODOLOGY FOR THE AUTOMATIC CREATION OF DIGITAL TWIN MOBILE NETWORKS APPLIED TO 5G ENVIRONMENTS

This section discusses the details of the proposed methodology for creating and deploying Digital Twins on the network.

After reviewing the documentation in sections II and III on technologies based on Digital Twins, we have identified the need to develop an automatic method to collect information to model the physical element, especially in the context of this work, the 5G networks. This method aims to create a mechanism for the automatic creation of networks Digital Twins that covers the entire lifecycle, from data collection, modeling, and deployment to the interconnection between the twins.

As can be seen in Figure 1, two states associated with the system state are defined, state 0 and state 1, which relate to data acquisition and data tracking, respectively. These two states condition the specific operation of the proposed methodology and allow to cover the different phases of the MNDT ecosystem.

- State 0: this is the initial phase. It is assumed that the MNDT ecosystem does not yet exist. In this phase, the proposed methodology focuses on data collection in the physical twin network with the main goal of modeling this network as a Network Digital Twin.
- State 1: once the methodology has been applied in state 0 (data acquisition), it is assumed that the NFV-based MNDT ecosystem has been created and the different twins are interconnected. At this stage, the methodology moves from capturing topological information for modeling to capturing monitoring information (state of the physical twins, traffic flows, network parameters, etc.) to provide the Digital Twin with data generated in real-time in its physical counterpart.

Once the two types of operations associated with the two possible states of the MNDT system (state 0 and state 1) are established, we can present the methodology in detail. In Figure 5 we can see that it consists of seven steps, executed one after the other, each of which is described below.

A. PHASE 1. DATA ACQUISITION

This is the initial phase of the methodology. The main objective of this phase is to obtain in an automated way topological information, information about hardware resources

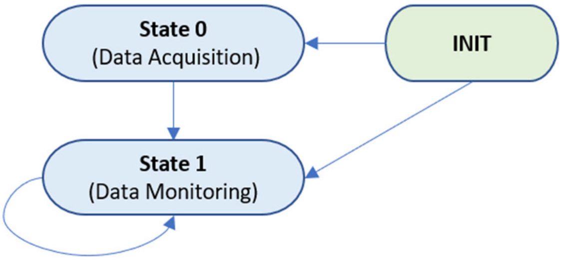


FIGURE 1. States associated with the proposed methodology.

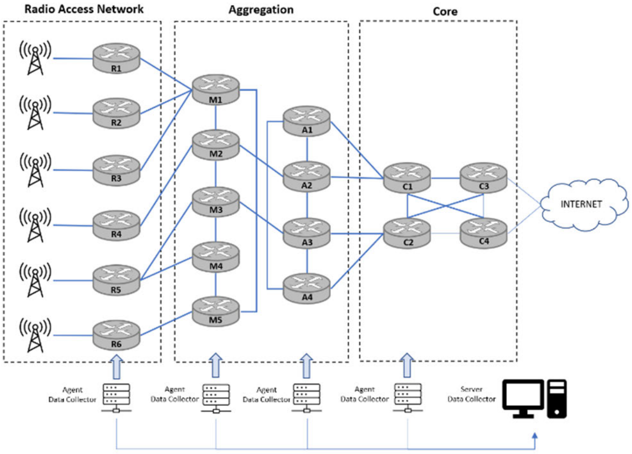


FIGURE 2. Phase 1. Data Acquisition example.

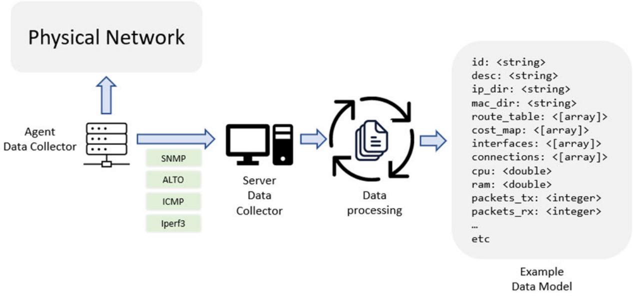
presents in the network, route tables, cost metrics between links, bandwidth measurements, and specific parameters of the elements that make up the network. To achieve an effective collection, the use of agents in the target physical network is proposed, which, in their state 0 (data collection), are constantly aware of the relevant parameters associated with this step as we can see in Figure 2. These agents collect this information using the protocols and tools described in Section II, such as ALTO, to obtain the topological map and the associated costs between the links, and SNMP to obtain specific parameters, such as the device descriptor, routing tables, the network interfaces present in the devices, Iperf3, to obtain measurements of the bandwidth between the links, etc. This information is sent to a central element responsible for receiving it and completing a data model whose goal is to semantically represent the topological, hardware, and software characteristics of the network selected as a physical twin.

This type of communication between the agents deployed in the network and the central management team can be decentralized, with the agents returning the information at specified intervals, or centralized, with the central team overseeing the ordering of the information-gathering tasks to each of the agents deployed in the target network.

At this point, it is assumed that the network is stabilized at the time of the start of the MNDT creation methodology, for which a finite period is specified to obtain the data.

B. PHASE 2. MODELING

Once phase 1 is complete, the information collected by the agents using various technologies and tools (ALTO, SNMP, ICMP, Iperf3, OpenFlow, etc.) is stored on the central computer. In the modeling phase, an in-depth analysis of the data obtained from the topology is performed, as we can see in Figure 3. In this process, correlation and data inference are applied to obtain a semantic representation of the network that includes as much detail as possible based on the data collected in the phase. 1. The aim of this phase is to obtain a simplified model of the physical network and to avoid a complete replication of the physical network as a Digital Twin, which would not be effective from the point of view of cost and resource optimization.



```
id: <string>
desc: <string>
ip_dir: <string>
mac_dir: <string>
route_table: <[array]>
cost_map: <[array]>
interfaces: <[array]>
connections: <[array]>
cpu: <double>
ram: <double>
packets_tx: <integer>
packets_rx: <integer>
…
etc
```

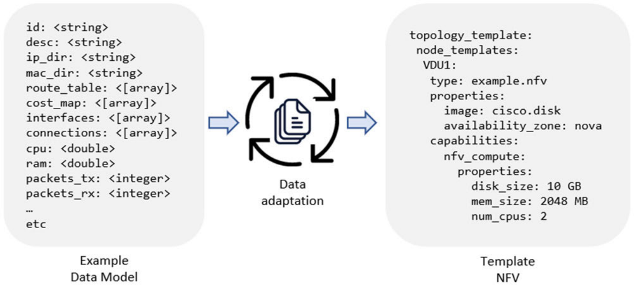
FIGURE 3. Phase 2. Data Modeling example.

C. PHASE 3. ADAPTATION

After the Digital Twin modeling phase, an adaptation phase is defined. Figure 4 shows a subset of information extracted from the data model about the physical parameters required to deploy the virtual resource in the NFV architecture that will host the network Digital Twin (hardware resources, software resources, network interfaces, network segments, etc.). As mentioned in the Background section, the use of proposed template technologies such as Jinja2, OASIS Tosca, YANG, etc. standardizes the deployment of Virtualization Deployment Units (VDUs) in the NFV architecture. In this phase, the data model obtained in phase 2 must be adapted to the template format compatible with NFV deployment. The remaining subset of information from the model can be used in Phase 5, which may cover the software deployment phase so that the Digital Twin behaves exactly like the physical twin.

D. PHASE 4. NFV DEPLOYMENT

After obtaining the various VDUs responsible for representing the twin physical elements as digital instances in the NFV architecture, depending on the simplification applied in the modeling phase, the MNDT setup phase begins. For this, the use of the NFV architecture proposed by ETSI is proposed, using the open-source project MANO, OSM, which fully covers the management and orchestration requirements of the NFV architecture.



```
id: <string>
desc: <string>
ip_dir: <string>
mac_dir: <string>
route_table: <[array]>
cost_map: <[array]>
interfaces: <[array]>
connections: <[array]>
cpu: <double>
ram: <double>
packets_tx: <integer>
packets_rx: <integer>
…
etc
```

```
topology_template:
 node_templates:
  VDU1:
   type: example.nfv
   properties:
     image: cisco.disk
     availability_zone: nova
   capabilities:
     nfv_compute:
       properties:
         disk_size: 10 GB
         mem_size: 2048 MB
         num_cpus: 2
```

FIGURE 4. Phase 3. Data Adaptation example.

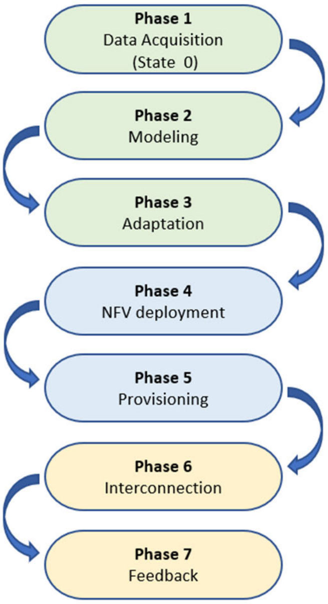


FIGURE 5. Methodology for Network Digital Twin associated to state 0.

As it will be seen in the next section, which covers the architecture proposal, OpenStack is set as the base technology for the VIM part of the NFV architecture. In addition, the multi-VIM configuration is capable of hosting VIMs based on container-based virtualization technologies such as Docker Swarm [53] or Kubernetes [54].

An important detail to note is that in this method, the 5G network core dedicated to signaling traffic is set up as a fixed element, both in the physical twin and in the Digital Twin. Therefore, the network information of the physical twin related to the 5G core is not collected in phase 1 or modeled in phase 2 but is set up as an element of the static MNDT that is provisioned and connected independently. to the MNDT model that models the topology of the target network.

E. PHASE 5. PROVISIONING

This phase of the methodology aims to complete and adapt specific features of the operation of the real network in the Digital Twin. At this stage, phase 4 must have completed the basic setup of the MNDT through the NFV architecture, both the MNDT model and the 5G core part discussed in the previous phase. In this phase, the use of the ANSIBLE [55] deployment automation tool is proposed, which would run specific playbooks for this MNDT to complete the characterization of the operation of the Digital Twin of the network based on the subset of information discussed in the adaptation phase.

F. PHASE 6. INTERCONNECTION

At this point, the MNDT is set up and deployed, but it still does not meet one of the key requirements of the Digital Twin technology, which is the connection between the physical twin and the Digital Twin. If this requirement is not met, we would only be dealing with a conventional network emulator that does not have real-time processing capabilities.

This phase presents two major challenges: the security of communication between the twins and the management of communication between the dual elements within each twin. Each of these challenges is discussed in more detail below:

- **Security**. Due to the nature of the physical elements being modeled in MNDT, it is critical to provide for the security of communication between the two worlds, physical and digital. In this phase of the methodology for the automatic creation of Digital Twins in the network, it is proposed the use of two devices that act as communication gateways, connecting and multiplexing the traffic between the two twins. These gateways must be located at the edge of the network in each one of the areas. Enabling the security of T2T (Twin 2 Twin) communication can be achieved mainly by using transport layer protocols such as TLS [56]. Specifically the PQC-based implementation offered by Open Quantum Safe through its liboqs library [57] and its implementation in OpenSSL [58] are recommended, due to its applicability to messaging protocols such as MQTT and Kafka.
- **Communication management.** To make the connection between the twins more granular, the use of differentiated information flows to connect the atomic elements of the physical network to their modeled counterparts in the Digital Twin is proposed. As mentioned in the modeling phase, the network model of the Digital Twin should ideally be less complex than its digital counterpart, so there will not necessarily be a correspondence of atomic elements within both twins. However, the information exchanged may relate to the final behavior of said elements, even if modeled. For this reason, the use of messaging-based flow management tools, such as MQTT [59], RabbitMQ [60], or Apache Kafka [61] is proposed, which would allow multiplexing of information, processing of independent data streams, and real-time communication between the two twins.

G. PHASE 7. FEEDBACK

The last phase of the proposed methodology consists in activating a feedback loop between the two twins, using the bidirectional connection introduced in the previous phase.

This phase completes state 0, associated with the automatic creation of the network's Digital Twins, and automatically transitions to state 1, in which the twins start monitoring tasks to exchange information to perform higher-level tasks, such as topology optimization, dataset acquisition, preventive actions in stressful situations in the network, etc.

The overview of the different phases that will be carried out first is part of the initial creation of the MNDT, which is associated with state 0. In it, the focus is on phase 1 of automated data acquisition to model the Digital Twin of the network. However, to fully cover the specific requirements and characteristics of the Digital Twin technology discussed in section II of this article, an additional state is required, which is achieved once the final phase of the methodology is in its initial start, or zero state.

The purpose of this state is to establish the feedback between the twins as introduced in phase 7. In Figure 6, we can see that in this state, phases 4 and 5, which are associated with deployment in the NFV architecture, are eliminated. However, in state one, the phases 2, 3, and 6 are retained with some minor changes, as the nature of the data collected changes.

The main idea is to reuse the agents that are deployed in the physical network and are responsible for collecting information and sending it to a central entity for processing, so that they have two different collection processes: In state 0, they collect information to characterize the network, and in state 1, they are responsible for collecting telemetry data that is used to operate the Digital Twin.

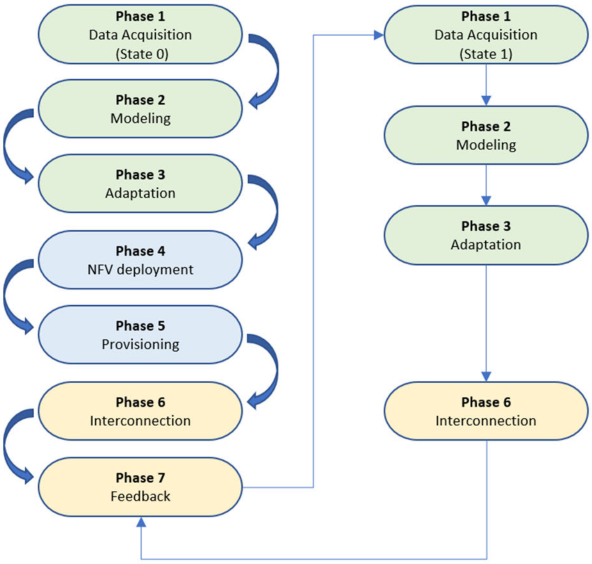


FIGURE 6. Methodology for Network Digital Twin associated to state 1.

This telemetry data or monitoring data can be very different and depends on the type of physical topology modeled. Therefore, even if phases 2 and 3 are reused, it is necessary to slightly change the operation of these phases in state 1.

- Phase 2 - Modeling (state 1). Depending on the type of information to be monitored, this phase can either model the information by adding data (e.g., information flows, network delays, etc.) based on the modeling performed for the MNDT, or it can redirect the information in pcaps format, in the case of traffic flows, directly into its digital counterpart.
- Phase 3 - Adaptation (state 1). In state 0, the adaptation phase involved mapping between the data model generated after information collection and the format of the templates or descriptors for creating VDUs in the NFV architecture. In state 1, it is not necessary to adapt to a specific data format since this information does not necessarily have to pass through the NFV architecture. It is only necessary to consider that the generated information format is compatible with the information format expected from the target element in the MNDT.

V. PROPOSED ARCHITECTURE

The proposed architecture is based on a modular and scalable system for the automated creation of a MNDT system applied to 5G networks. Additionally, the methodology proposed in Section IV is adapted to this architecture to model each element present in a 5G network as a Digital Twin. This creates a complete MNDT model that accurately emulates the behavior of a 5G network in specific use or study cases, utilizing the MNDT to perform various tasks based on technologies such as artificial intelligence. Figure 8 shows the conceptual design of the overall system. In the following section, we discuss the various modules of the MNDT architecture as shown in Figure 7.

A. MNDT SYSTEM INPUT

This is the first module of the platform for the use of the MNDT. This module incorporates the information collected, processed, and adapted in phases 1, 2 and 3 of the proposed methodology.

Basically, a set of software agents is deployed in the physical architecture that, through a software implementation based on the ALTO protocol, ICMP polls, SNMP queries and measurements with the Iperf3 tool, collect all the information necessary to obtain a semantic representation of the network, as well as additional information that facilitates the task of modeling the MNDT. The information reported by the agents is centralized in a server computer that is responsible for processing, inferring, and adapting the information to the data model to be used in the MNDT.

At this point, it is assumed that the operator's network is stable, i.e., all routing information is converged, it is bounded, and administrator-level permissions are in place to connect these agents to the network.

This task corresponds to state 0 of the proposed methodology, where the MNDT has not yet been created on the NFV architecture proposed by ETSI. Once stage 3 of the methodology is achieved, the physical network topology will

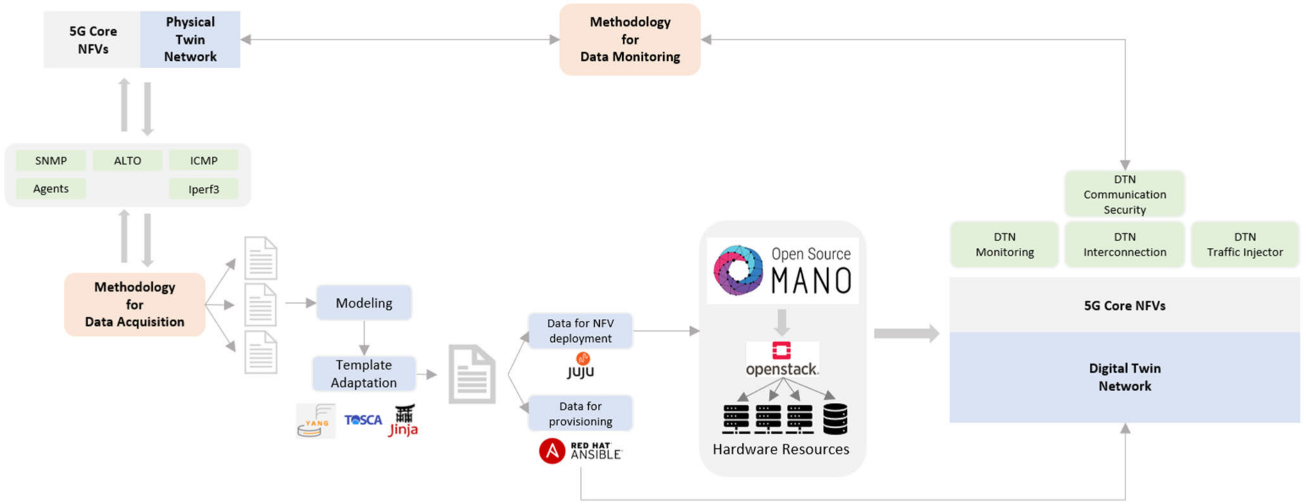


FIGURE 7. Architecture proposal for the creation and automatic deployment of Network Digital Twins.

be modeled and adapted to the template formats required for the VNFD and the NSD will be introduced into the NFV platform implemented through Open-Source MANO to leverage NFV.

B. NFV DEPLOYMENT MODULE BASED ON OpenMANO AND OpenStack

Once the initial information-gathering phase for the creation of the MNDT is complete, the deployment module will be used, based on the NFV architecture along with OpenMANO and using OpenStack technology as VIM (Virtual Infrastructure Manager) systems that will oversee the deployment of the virtual infrastructure that will host the MNDT in VNF format. One of the advantages of deploying using these technologies is the ability to slice the 5G network, which allows for the creation of multiple MNDT deployments that are isolated from each other. This deployment module uses a catalog of images and containers that serve as the basis for deploying and configuring the various MNDT, which are implemented using technologies such as Kubernetes or Docker.

For the specific application of MNDT to 5G networks, we propose using the 5G core network VNFs described in Section II. These VNFs can be deployed either in a single virtual instance with a monolithic architecture, or in a distributed manner that enables communication between the different VNFs. To apply a real-world context within the MNDT 5G core, we proposed the inclusion of testing devices, both virtual and hardware, that act as clients or servers to emulate different types of traffic, such as streaming video or web pages. Additionally, an orchestrator would be deployed to communicate with all the devices that comprise the MNDT through a management network. The orchestrator would be capable of monitoring and capturing all traffic, performing analytics, compiling statistics, generating reports, and creating data sets for further use by AI algorithms.

C. COMMUNICATION AND INTERCONNECTION BETWEEN TWINS

Once the MNDT infrastructure has been established and deployed, the various lines to be used throughout the system must be set up. Depending on the type of interconnected elements and their communication needs, two different lines or pipelines are distinguished: V2V (Virtual to Virtual) and P2V (Physical to Virtual). The first one is executed within the MNDT and connects the individual DTs that make up the entire system. These pipelines allow us to mirror the communication behavior in the physical world, without the time limitation that exists there. Thus, large data streams can be communicated in a shorter time, facilitating emulation, and speeding up results. The second enables the connection between the virtual world of the MNDT and the physical world, creating a continuous feedback loop between the MNDT and its physical counterpart, providing the system with continuous coevolution and cooperation capability.

Taking advantage of the elements used for data acquisition, the use of the same architecture is proposed to establish the connection between the two twins. The computers that act as servers to collect the information reported by the agents of both twins, in turn, act as communication gateways that allow the application of mechanisms for encrypting the traffic to ensure the security of the system. The information exchanged between the two gateways is in turn demultiplexed into communication flows based on messaging protocols such as MQTT [59], Kafka [61] or RabbitMQ [60], so that a direct relationship can be established between each specific element of the physical twin and its digital counterpart as shown in Figure 8.

D. MONITORING MODULE

This module is responsible for monitoring the overall operation of all information exchanged within the MNDT. Once all

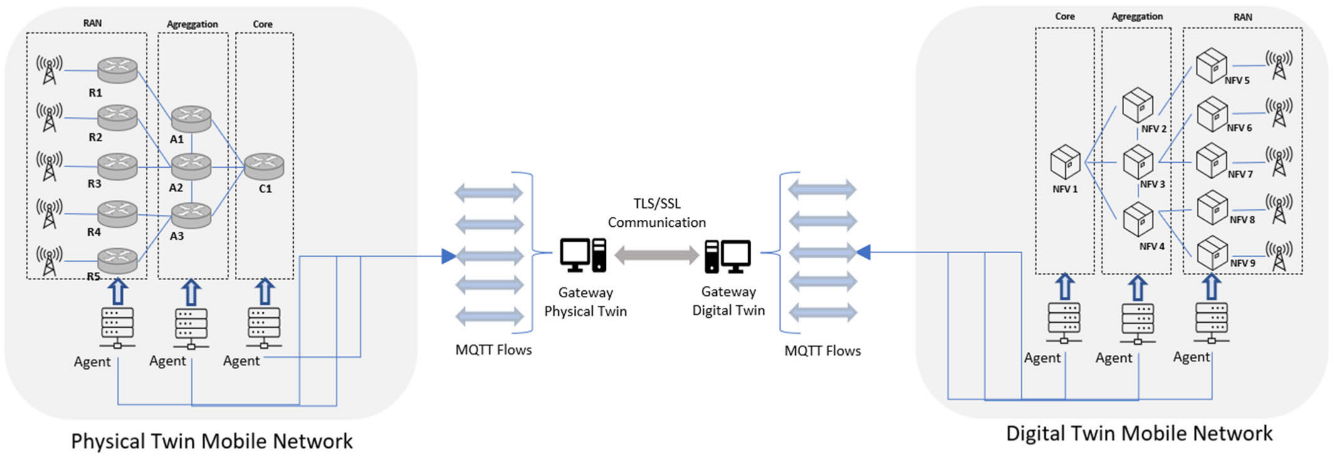


FIGURE 8. Interconnection between twins.

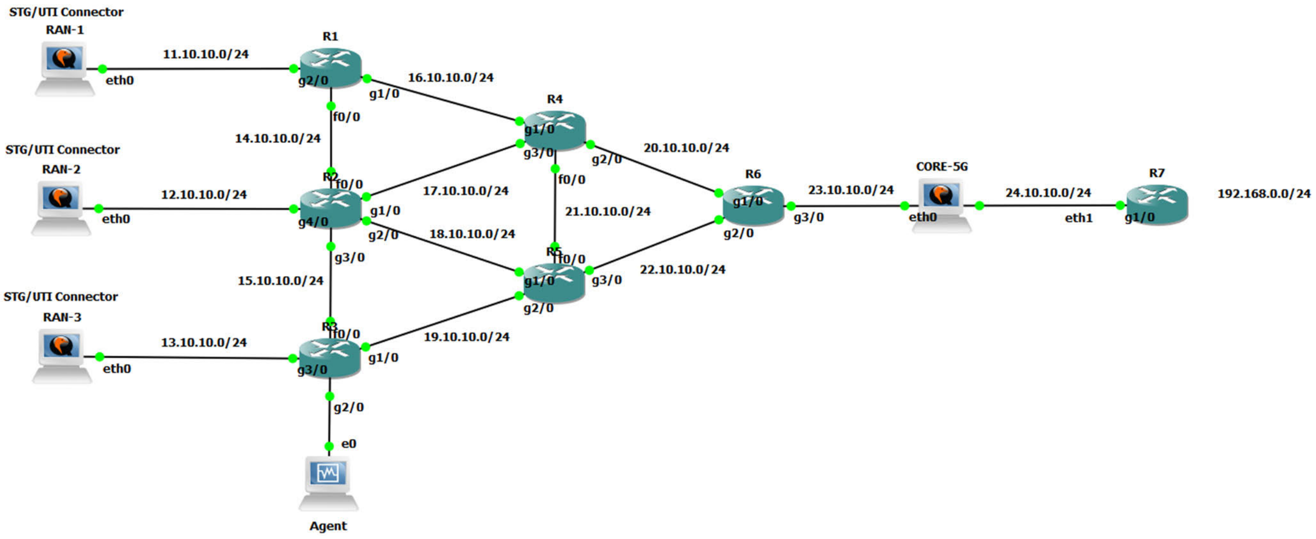


FIGURE 9. Testbed environment based GNS3 using STG/UTI connector and Core 5G based on free5gc.

twins are provisioned and configured, the monitoring module controls the activation of port mirroring functions in each of the subnets present in the MNDT to use this information traffic for the generation of data sets that allow the MNDT's emulation.

E. TRAFFIC INJECTION MODULE

In addition, it is envisioned to enable traffic generation or injection based on different 5G traffic models over the MNDT, allowing validation and analysis of network performance. The main objective of this module is to provide a method for injecting traffic into the virtualized infrastructure of the 5G core network without having to deal with real radio access networks and 5G hardware. This module acts as a signaling NAS traffic generator that can communicate with the VNF AMF in the 5G core network and emulate UE operation in a real environment. This module can perform session management, registration, UE registration, etc. It can also maintain GTP tunnels with the VNF UPF of the core network and send data over them. User data sent through this module, which acts as an intermediary, is captured on a network interface, and can originate from any virtual device or hardware machine.

Both the signaling emulation NAS (Signaling Traffic Generator - STG) and the user traffic injector (User Traffic Injector - UTI) [62], [63], [64] are implemented as a single process. This enables the registration of end devices, the establishment of PDU sessions and the sending of user data to a network with the same software tool that connects the signaling data to the tunnel generated for the user traffic.

VI. TESTBED ENVIRONMENT

A prototype has been developed for the implementation of the proposed methodology, focusing on the initial phases, to obtain a semantic representation of the network topology and its adaptation to NFV platform.

We have developed an agent that implements the initial phases of the proposed methodology and integrates them into a test scenario provided with GNS3, as shown in Figure 9. This scenario represents a physical network using 5G Technology in Stand-Alone mode. This scenario is about replicating an Industry 4.0 use case where different sites with independent 5G areas are connected to a 5G core implementation via an aggregation network based on Cisco devices.

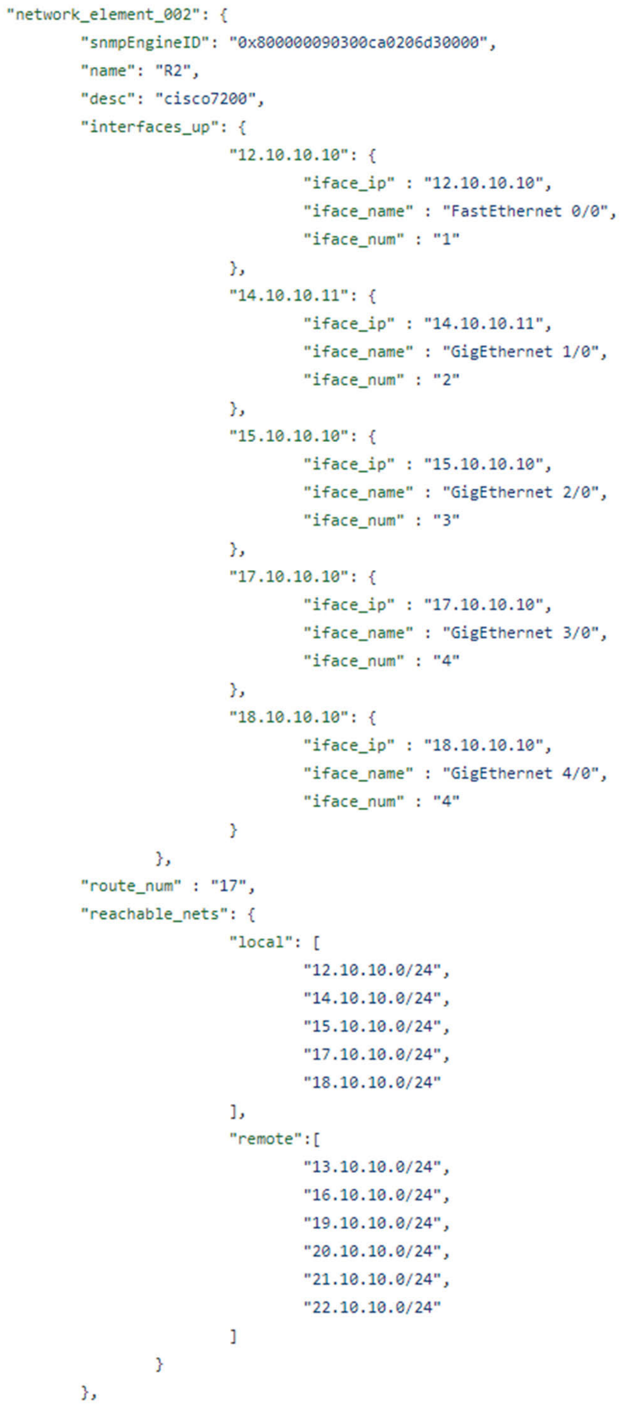
To implement a 5G Stand-Alone scenario, we used two key elements. For the first element, the 5G core, we chose the Free5GC implementation because of its maturity and Open-Source philosophy. The second is a tool capable of injecting 5G traffic into the network, STG/UTI [64] module as we can see in the previous section. Although it is not necessary to have this type of module for the application of the first phases of the methodology, it is important to have this capacity once the deployment and interconnection of the twins has taken place.

The developed agent, responsible for the execution of the methodology's first phases, has been developed with the main objective of being completely autonomous. It is only necessary to connect the agent to the aggregation network for it to start exploring the network to determine all the information about the network topology (devices, subnets, hops, connections, etc.). Two common types of networks have been identified in agent development, traditional network (purely hardware-based) and software defined networks. This agent will initially cover the first type of network, using protocols such as SNMP, ICMP, RIP, OSPF, etc. The second type of network will allow the use of OpenFlow to detect the topology in SDN networks.

VII. EVALUATION

Using the testbed environment described in the previous section we have carried out a series of tests to validate the first phases of the proposed methodology.

First, the agent connects to one of the routers in the aggregation network, the agent will perform a self-discovery of its next-hop router, first making basic SNMP queries to obtain parameters such as the unique device identifier, the name, the type of interfaces available on the computer, the IP addresses assigned to these interfaces, the number of routes established in the router, and all the network segments reachable from this computer. Once this information is collected, a data model in JSON format is created. This model stores the information of the network element that has already been analyzed. Then, the process is repeated in the new subnets discovered based on the IPs assigned to the interfaces. We assume that the aggregation network consists only of network devices where SNMP is active. Once this investigation is done, the new SNMP elements are identified. network and the algorithm developed in the agent continues to traverse the network architecture.



```
"network_element_002": {
        "snmpEngineID": "0x800000090300ca0206d30000",
        "name": "R2",
        "desc": "cisco7200",
        "interfaces_up": {
                        "12.10.10.10": {
                                "iface_ip" : "12.10.10.10",
                                "iface_name" : "FastEthernet 0/0",
                                "iface_num" : "1"
                        },
                        "14.10.10.11": {
                                "iface_ip" : "14.10.10.11",
                                "iface_name" : "GigEthernet 1/0",
                                "iface_num" : "2"
                        },
                        "15.10.10.10": {
                                "iface_ip" : "15.10.10.10",
                                "iface_name" : "GigEthernet 2/0",
                                "iface_num" : "3"
                        },
                        "17.10.10.10": {
                                "iface_ip" : "17.10.10.10",
                                "iface_name" : "GigEthernet 3/0",
                                "iface_num" : "4"
                        },
                        "18.10.10.10": {
                                "iface_ip" : "18.10.10.10",
                                "iface_name" : "GigEthernet 4/0",
                                "iface_num" : "4"
                        }
                },
        "route_num" : "17",
        "reachable_nets": {
                        "local": [
                                "12.10.10.0/24",
                                "14.10.10.0/24",
                                "15.10.10.0/24",
                                "17.10.10.0/24",
                                "18.10.10.0/24"
                        ],
                        "remote":[
                                "13.10.10.0/24",
                                "16.10.10.0/24",
                                "19.10.10.0/24",
                                "20.10.10.0/24",
                                "21.10.10.0/24",
                                "22.10.10.0/24"
                        ]
                }
        },
```

FIGURE 10. Example of the JSON representation of the R3 aggregation network element.

In this implementation, it is assumed that each element of the aggregation network is able to reach the rest of the computers. Thus, if we know the number of subnets reachable by a router, we achieve convergence of the algorithm. However, by storing the unique SNMP identifier, snmpEngineID, we avoid processing loops when analyzing a router that is connected to other routers already processed by the algorithm.

Once convergence is achieved, a file is generated in JSON format, as shown in the example in Figure 10, describing the entire network with relevant information for modeling in the form of a VNF for each network element on stage.

At this stage, the agent covers stages one and two of the proposed methodology associated with state zero, but the digital twin has not yet been created. After receiving the JSON file, the agent performs the adaptation tasks associated with stage three of the methodology, where it models each network element into VNF functions using TOSCA templates compatible with the NFV architecture proposed by ETSI. As a result, N VNFD files are obtained, which model each network element, as well as the Virtual Link Descriptor (VLD) files required to connect all VNFs in the Digital Twin deployment phase.

It has been shown that the developed agent, which currently implements the first three phases of the proposed methodology, obtains the minimum information necessary to begin the establishment of the network in the form of a digital twin on an NFV architecture. In the further developments of the implementation, the focus will be on the determination of the parameters for the use of the devices (CPU and RAM) that will allow, in phase four, to correctly adjust the resources allocated to the DT in VNF format and to avoid the oversizing of the resources in the MNDT.

After completing phase three of the methodology, in which all elements of the 5G aggregation network have been modeled in VNF format, we proceed with deployment on the NFV architecture according to phase four of the methodology. In this case, the NFV deployment is on an architecture based on Open-Source MANO and using OpenStack as VIM to perform the deployment of the VNFs. At this point, only the deployment at the topological level is addressed. The interconnected VNFs are provisioned according to the information collected in phase one of the topological characterization of the network.

VIII. CONCLUSION

While the Digital Twin paradigm, particularly Network Digital Twins, is being explored in various publications, the implementation of some of its key features in mobile communication networks area remains unclear. This paper presents a specific methodology for the development and implementation of MNDT 5G platforms, which is directly aligned with the NFV architecture proposed by ETSI, and a global platform reflecting the whole MNDT ecosystem. The methodology covers the main features of the Digital Twin paradigm identified in the literature and focuses on the phases of data acquisition for subsequent modeling, as well as the interconnection between physical and digital twins.

Main contributions made in this work are:

- Design of a Methodology to cover the entire life cycle associated with digital twin-based development, with a specific focus on 5G mobile networks.
- Deployment of smart agents responsible for collecting the minimum information necessary for the topological representation of the network, as well as additional information to facilitate data-driven modeling for the creation of the digital twin.
- Structuring the collected data to obtain a data model capable of being represented and deployed on NFV architectures.
- Identification of the technical needs associated with digital twin technology, specifically covered in the interconnection phase of the proposed methodology, which allows connections between digital twins and their physical counterparts.
- Initial approach to the security model to be used in this type of architecture, with an emphasis on securing communications between twins and proposing the evolution towards the use of post-quantum cryptography.
- Establishment of a technological framework that would enable the deployment of any target network in digital twin format.

Although the work proposed in this article is still in its initial phase, it has demonstrated that the development presented in the previous sections is well-suited to the needs and technologies of the Digital Twins paradigm. Specifically, the proposed methodology is applied to mobile networks, facilitating the modeling tasks of this type of network and providing a method for the semantic representation of the topology and its adaptation to NFV technology. We believe that the approach outlined in this article, in alignment with the work carried out by the IETF, as well as the selection of technologies for the MNDT deployment platform's structure, has the potential to be highly beneficial in the development and evolution of new digital twin-based systems in the specific context of next-generation networks.

However, there is room for improvement in this work, certain needs were identified during the implementation phase that must be addressed in subsequent steps. These needs include:

- The topological characterization of SDN-based networks using protocols such as OpenFlow, so that the developed agent can collect topological information regardless of the network type.
- The development of subsequent of the methodology, with a particular emphasis on validating the interconnection of digital twins based on messaging protocols with support for post-quantum cryptography.
- The ability to dynamically reconfigure the MNDT based on optimization decisions for one of the twins, so that the methodology can switch between 0 and 1 states to characterize and reconfigure either the physical or digital twin when either makes a change that affects the topology or scaling of one of its elements, with the goal of optimizing the data flows that support 5G networks.

REFERENCES

[1] Y. Wu, K. Zhang, and Y. Zhang, "Digital twin networks: A survey," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 13789–13804, Sep. 2021, doi: 10.1109/JIOT.2021.3079510.

[2] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, ''Digital twin for 5G and beyond,'' *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 10–15, Feb. 2021, doi: 10.1109/MCOM.001.2000343.

[3] *Draft Concepts of Digital Twin Network*. Accessed: Dec. 1, 2022. [Online]. Available: https://datatracker.ietf.org/doc/id/draft-zhou-nmrg-digitaltwin-network-concepts-03.txt

[4] S. Vakaruk, A. Mozo, A. Pastor, and D. R. López, ''A digital twin network for security training in 5G industrial environments,'' in *Proc. IEEE 1st Int. Conf. Digit. Twins Parallel Intell. (DTPI)*, Aug. 2021, pp. 395–398, doi: 10.1109/DTPI52967.2021.9540146.

[5] A. Pastor, A. Mozo, D. R. Lopez, J. Folgueira, and A. Kapodistria, ''The mouseworld, a security traffic analysis lab based on NFV/SDN,'' in *Proc. 13th Int. Conf. Availability, Rel. Secur.*, Aug. 2018, pp. 1–6, doi: 10.1145/3230833.3233283.

[6] C. Zhou, H. Yang, X. Duan, D. Lopez, A. Pastor, Q. Wu, M. Boucadair, and C. Jacquenet. (Oct. 2022). *Digital Twin Network: Concepts and Reference Architecture*. Accessed: Dec. 1, 2022. [Online]. Available: https://datatracker.ietf.org/doc/draft-irtf-nmrg-network-digital-twin-arch

[7] M. W. Grieves and M. Tanniru, ''PLM, process, practice and provenance: Knowledge provenance in support of business practices in product lifecycle management,'' *Int. J. Product Lifecycle Manage.*, vol. 3, no. 1, p. 37, 2008, doi: 10.1504/IJPLM.2008.019969.

[8] E. J. Tuegel, A. R. Ingraffea, T. G. Eason, and S. M. Spottswood, ''Reengineering aircraft structural life prediction using a digital twin,'' *Int. J. Aerosp. Eng.*, vol. 2011, pp. 1–14, Jan. 2011, doi: 10.1155/2011/154798.

[9] S. Mihai, M. Yaqoob, D. V. Hung, W. Davis, P. Towakel, M. Raza, and M. Karamanoglu, ''Digital twins: A survey on enabling technologies, challenges, trends and future prospects,'' *IEEE Commun. Surveys Tuts.*, vol. 24, no. 4, pp. 2255–2291, 4th Quart., 2022, doi: 10.1109/COMST.2022.3208773.

[10] 3GPP. *Release 16*. Accessed: Dec. 1, 2022. [Online]. Available: https://www.3gpp.org/specifications-technologies/releases/release-16

[11] 3GPP. *Release 17*. Accessed: Dec. 1, 2022. [Online]. Available: https://www.3gpp.org/specifications-technologies/releases/release-17

[12] *IMT Vision—Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond*, document Recommendation ITU-R M.2083-0, ITU, Sep. 2015. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2083-0-201509-I!!PDF-E.pdf

[13] D. Kreutz, F. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, ''Software-defined networking: A comprehensive survey,'' *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015, doi: 10.1109/JPROC.2014.2371999.

[14] C. Bouras, A. Kollia, and A. Papazois, ''SDN & NFV in 5G: Advancements and challenges,'' in *Proc. 20th Conf. Innov. Clouds, Internet Netw. (ICIN)*, Mar. 2017, pp. 107–111, doi: 10.1109/ICIN.2017.7899398.

[15] *What is Network Slicing?* Accessed: Dec. 1, 2022. [Online]. Available: https://5g.co.uk/guides/what-is-network-slicing/

[16] Devopedia. (Mar. 26, 2021). *5G Service-Based Architecture*. Accessed: Dec. 1, 2022. [Online]. Available: https://devopedia.org/5g-service-based-architecture

[17] (Oct. 31, 2022). *Free5GRAN*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/free5G/free5GRAN

[18] GitLab. *OAI/Openairinterface5G? GitLab*. Accessed: Dec. 1, 2022. [Online]. Available: https://gitlab.eurecom.fr/oai/openairinterface5g

[19] (Dec. 1, 2022). *srsRAN*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/srsran/srsRAN

[20] A. Güngör. (Dec. 1, 2022). *Aligungr/UERANSIM*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/aligungr/UERANSIM

[21] (Dec. 1, 2022). *free5GC/free5GC*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/free5gc/free5gc

[22] F. Okuyucu. (Nov. 25, 2022). *Internship-5GCN*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/bubblecounter/Internship-5GCN

[23] GitLab. *CN5G · GitLab*. Accessed: Dec. 1, 2022. [Online]. Available: https://gitlab.eurecom.fr/oai/cn5g

[24] Open5GS. (Dec. 1, 2022). *open5GS/open5GS*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/open5gs/open5gs

[25] *GitHub—Miaoski/Docker_open5GS: Docker Files to Run open5GS in a Docker*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/miaoski/docker_open5gs

[26] (Nov. 28, 2022). *Free5gmano NM Manager*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/free5gmano/free5gmano

[27] Orange. (Nov. 9, 2022). *Towards5GS-Helm*. Accessed: Dec. 1, 2022. [Online]. Available: https://github.com/Orange-OpenSource/towards5gs-helm

[28] ETSI. *ETSI—Welcome to the World of Standards!* Accessed: Dec. 1, 2022. [Online]. Available: https://www.etsi.org/

[29] M. Veeraraghavan, T. Sato, M. Buchanan, R. Rahimi, S. Okamoto, and N. Yamanaka, ''Network function virtualization: A survey,'' *IEICE Trans. Commun.*, vol. 100, no. 11, 2017, Art. no. 2016NNI0001, doi: 10.1587/transcom.2016NNI0001.

[30] *11. ANNEX 3: OSM Information Model—Open Source MANO 6.0 Documentation*. Accessed: Dec. 1, 2022. [Online]. Available: https://osm.etsi.org/docs/user-guide/latest/11-osm-im.html

[31] *OSM Information Model—OSM Public Wiki*. Accessed: Dec. 1, 2022. [Online]. Available: https://osm.etsi.org/wikipub/index.php/OSM_Information_Model

[32] *TOSCA Simple Profile for Network Functions Virtualization (NFV)-Version 1.0*. Accessed: Dec. 1, 2022. [Online]. Available: https://docs.oasis-open.org/tosca/tosca-nfv/v1.0/tosca-nfv-v1.0.html

[33] N. E. H. Nouar, S. Yangui, N. Faci, K. Drira, and S. Tazi, ''A semantic virtualized network functions description and discovery model,'' *Comput. Netw.*, vol. 195, Aug. 2021, Art. no. 108152, doi: 10.1016/j.comnet.2021.108152.

[34] K. Kaur, V. Mangat, and K. Kumar, ''A review on virtualized infrastructure managers with management and orchestration features in NFV architecture,'' *Comput. Netw.*, vol. 217, Nov. 2022, Art. no. 109281, doi: 10.1016/j.comnet.2022.109281.

[35] Pallets. *Jinja*. Accessed: Dec. 1, 2022. [Online]. Available: https://palletsprojects.com/p/jinja/

[36] *NFV Descriptos Based on YANG Specification*. Accessed: Jan. 14, 2023. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/NFV-SOL/001_099/006/02.06.01_60/gs_NFV-SOL006v020601p.pdf

[37] *YANG—ETSI Code*. Accessed: Jan. 14, 2023. [Online]. Available: https://forge.etsi.org/rep/nfv/SOL006/tree/v2.6.1

[38] *YANG—Oficial Repo*. Accessed: Jan. 14, 2023. [Online]. Available: https://github.com/YangModels/yang

[39] *RFC 7950*. Accessed: Jan. 14, 2023. [Online]. Available: https://datatracker.ietf.org/doc/rfc7950/

[40] M. Segovia and J. Garcia-Alfaro, ''Design, modeling and implementation of digital twins,'' *Sensors*, vol. 22, no. 14, p. 5396, Jul. 2022, doi: 10.3390/s22145396.

[41] *Eclipse Ditto^{TM} • Open Source Framework for Digital Twins in the IoT*. Accessed: Mar. 14, 2023. [Online]. Available: https://www.eclipse.org/ditto/

[42] (Feb. 2, 2021). *FIWARE—Open APIs for Open Minds*. Accessed: Mar. 14, 2023. [Online]. Available: https://www.fiware.org/

[43] A. Albo, L. Svedlund, and P. Falkman, ''Modular virtual preparation method of production systems using a digital twin architecture,'' in *Proc. 26th IEEE Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2021, pp. 1–8, doi: 10.1109/ETFA45728.2021.9613654.

[44] P. Almasan, M. Ferriol-Galmés, J. Paillisse, J. Suárez-Varela, D. Perino, D. López, A. A. P. Perales, P. Harvey, L. Ciavaglia, L. Wong, V. Ram, S. Xiao, X. Shi, X. Cheng, A. Cabellos-Aparicio, and P. Barlet-Ros, ''Digital twin network: Opportunities and challenges,'' 2022, *arXiv:2201.01144*.

[45] Y. Zhu, D. Chen, C. Zhou, L. Lu, and X. Duan, ''A knowledge graph based construction method for digital twin network,'' in *Proc. IEEE 1st Int. Conf. Digit. Twins Parallel Intell. (DTPI)*, Aug. 2021, pp. 362–365, doi: 10.1109/DTPI52967.2021.9540177.

[46] H. Wang, Y. Wu, G. Min, and W. Miao, ''A graph neural network-based digital twin for network slicing management,'' *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1367–1376, Feb. 2022, doi: 10.1109/TII.2020.3047843.

[47] W. Sun, H. Zhang, R. Wang, and Y. Zhang, ''Reducing offloading latency for digital twin edge networks in 6G,'' *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12240–12251, Aug. 2020, doi: 10.1109/TVT.2020.3018817.

[48] G. N. Schroeder, C. Steinmetz, R. N. Rodrigues, R. V. B. Henriques, A. Rettberg, and C. E. Pereira, ''A methodology for digital twin modeling and deployment for industry 4.0,'' *Proc. IEEE*, vol. 109, no. 4, pp. 556–567, Apr. 2021, doi: 10.1109/JPROC.2020.3032444.

[49] P. Janda, Z. Hajicek, and P. Bernardin, ''Implementation of the digital twin methodology,'' in *Proc. DAAAM*, vol. 1, 1st ed., 2019, pp. 533–538, doi: 10.2507/30th.daaam.proceedings.072.

[50] C. Gehrmann and M. Gunnarsson, ''A digital twin based industrial automation and control system security architecture,'' *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 669–680, Jan. 2020, doi: 10.1109/TII.2019.2938885.

[51] C.-C. Chung, C.-C. Pai, F.-S. Ching, C. Wang, and L.-J. Chen, "When post-quantum cryptography meets the Internet of Things: An empirical study," in *Proc. 20th Annu. Int. Conf. Mobile Syst., Appl. Services*, Jun. 2022, pp. 525–526, doi: 10.1145/3498361.3538766.

[52] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, pp. 188–194, Sep. 2017, doi: 10.1038/nature23461.

[53] Docker Documentation. (Nov. 30, 2022). *Swarm Mode Overview*. Accessed: Dec. 1, 2022. [Online]. Available: https://docs.docker.com/engine/swarm/

[54] Kubernetes. *Production-Grade Container Orchestration*. Accessed: Dec. 1, 2022. [Online]. Available: https://kubernetes.io/

[55] A. H. Red. *Ansible is Simple IT Automation*. Accessed: Dec. 1, 2022. [Online]. Available: https://www.ansible.com

[56] E. Rescorla and T. Dierks, *The Transport Layer Security (TLS) Protocol Version 1.2*, document RFC 5246, Internet Engineering Task Force, Aug. 2008, doi: 10.17487/RFC5246.

[57] Open Quantum Safe. (Mar. 14, 2023). *LibOQS*. Accessed: Mar. 14, 2023. [Online]. Available: https://github.com/open-quantum-safe/liboqs

[58] Open Quantum Safe. (Mar. 6, 2023). *OQS-OpenSSL_1_1_1*. Accessed: Mar. 14, 2023. [Online]. Available: https://github.com/open-quantum-safe/openssl

[59] *MQTT—The Standard for IoT Messaging*. Accessed: Dec. 1, 2022. [Online]. Available: https://mqtt.org/

[60] *Messaging That Just Works—RabbitMQ*. Accessed: Dec. 1, 2022. [Online]. Available: https://www.rabbitmq.com/

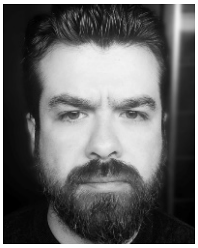
[61] *Apache Kafka*. Accessed: Dec. 1, 2022. [Online]. Available: https://kafka.apache.org/

[62] F. Rebecchi, A. Pastor, A. Mozo, C. Lombardo, R. Bruschi, I. Aliferis, R. Doriguzzi-Corin, P. Gouvas, A. Alvarez Romero, A. Angelogianni, I. Politis, and C. Xenakis, "A digital twin for the 5G era: The SPIDER cyber range," in *Proc. IEEE 23rd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2022, pp. 567–572, doi: 10.1109/WoWMoM54355.2022.00088.

[63] *GitHub—UPM-RSTI/STGUTG: Signaling and User Traffic Injection Tools for 5G Testing Platforms*. Accessed: Dec. 9, 2022. [Online]. Available: https://github.com/UPM-RSTI/STGUTG

[64] D. Rivera, J. I. Moreno, M. S. Rodrigo, D. R. López, and A. Mozo, "Providing heterogeneous signaling and user traffic for 5G core network functional testing," *IEEE Access*, vol. 11, pp. 2968–2980, 2023, doi: 10.1109/ACCESS.2022.3233412.

MARIO SANZ RODRIGO was born in Madrid, Spain, in 1989. He received the degree in communication systems engineering from the Carlos III University of Madrid (UC3M), Spain, in 2017, and the M.Sc. degree in cybersecurity from Universidad Politécnica de Madrid (UPM), Spain, in 2019. He is currently a Researcher of telematics engineering with UPM. He has been involved in European and national projects related with video coding, network monitoring, microservice virtualization, telemetry in electrical network with PLC using PRIME protocol, security in industrial environments, and SCADA systems.



DIEGO RIVERA received the B.S. degree in computer science and the M.S. and Ph.D. degrees in information technologies and communications from Universidad de Alcala, Spain, in 2010, 2013, and 2019, respectively. He is currently an Associate Professor and a Senior Researcher with the Research Group on Telecommunication and Internet Networks and Services (RSTI), Universidad Politécnica de Madrid, Spain. Since 2010, he has been working in several nationally and internationally funded research projects and has coauthored more than 40 publications, including research articles, book chapters, conference contributions, and patents. His research interests include computer network architectures and protocols, the Internet of Things architectures, and cybersecurity systems.

JOSÉ IGNACIO MORENO (Senior Member, IEEE) received the Ph.D. degree in telecommunication from Universidad Politécnica de Madrid (UPM), in 1996. He is currently a Full Professor with UPM. Since 1992, he has been working with international research projects related with protocol designs, protocol engineering, network management, advanced networks, and wireless system performance. During the last 15 years, he has led national and European projects on ICT topics with special focus on WSN, smart grids, and 5G technologies. He has published more than 100 papers in the field of advanced communications in technical books, magazines, and conferences.

MANUEL ÁLVAREZ-CAMPANA received the Telecommunication Engineering degree and the Ph.D. degree in telecommunication engineering from Universidad Politécnica de Madrid (UPM), in 1989 and 1995, respectively. He is currently an Associate Professor with UPM. He has participated in several projects within the framework of European and Spanish public funded research and development programs and in consultancy contracts for private companies and public organizations. His professional interests cover a broad spectrum of aspects related to communication networks and services. He is the author of more than 70 technical publications in journals and conferences and three books. His research interests include mobile communication networks, cybersecurity, and the Internet of Things.

DIEGO R. LÓPEZ joined Telefonica I+D, in 2011, as a Senior Technology Expert, where he is currently an in charge of the technology exploration activities with the GCTIO Unit. Before joining Telefonica, he spent some years in the academic sector, dedicated to research on network services, and was appointed as a member of the High-Level Expert Group on Scientific Data Infrastructures by the European Commission. His research interests include network infrastructures, with a special emphasis on virtualization, data-driven management, new architectures, and security. He is an ETSI Fellow and chairs the ETSI ISGs ZSM (on network automation) and PDL (on distribute ledger technologies). He is the NOC of ETSI ISG NFV.

• • •