IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 4 November 2022, accepted 21 December 2022, date of publication 30 December 2022,
date of current version 11 January 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3233412

APPLIED RESEARCH

Providing Heterogeneous Signaling and User Traffic for 5G Core Network Functional Testing

DIEGO RIVERA[1], JOSÉ IGNACIO MORENO[1], (Senior Member, IEEE), MARIO SANZ RODRIGO[1], DIEGO R. LÓPEZ[2], AND ALBERTO MOZO[3]
[1]Departamento de Ingeniería de Sistemas Telemáticos, ETSI de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]Telefónica I+D, 28010 Madrid, Spain
[3]Departamento de Sistemas Informáticos, ETSI de Sistemas Informáticos, Universidad Politécnica de Madrid, 28031 Madrid, Spain

Corresponding author: Diego Rivera (diego.rivera@upm.es)

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under Grant 833685 [a cybersecurity Platform for virtualiseD 5G cybEr Range services (SPIDER)], and in part by the Spanish Ministerio de Asuntos Económicos y Transformación Digital UNiversalización de Infraestructuras digitales para la COhesión-5G, Investigación + Desarrollo (UNICO-5G I+D) Program through the Project Beyond 5G Enhanced Management through dIgital twiNs based on artificial Intelligence (B5GEMINI) under Grant TSI-063000-2021-81 and Grant TSI-063000-2021-79.

ABSTRACT In the next 10 years, the telecommunications sector will be dominated by the deployment of pervasive networking available in a wide range of heterogeneous scenarios applying 5G (and beyond) technologies. SDN, NFV as well as Network Sliding Technologies used by 5G will provide a ‘‘softwarization’’ of the network functional elements, opening the ecosystem of network providers to a vast number of suppliers from traditional manufacturers to start-ups focusing on specialized functions. Due to the need to assert the expected interoperability of the introduction and evolution of this set of Network Functions, a set of validation tools is needed to explore traffic patterns according to different scenarios and thus allow the performance of the software modules to be evaluated. This research work presents the design of software tools that aim to provide a way for generating signaling and user traffic in 5G networks, enabling the definition of novel pure-software testing environments for 5G that includes real signaling and user traffic. The tools have been designed to support a set of testing signaling procedures that cover the basic main functioning of 5G networks and that allow for the transmission of user traffic over the network. The tools have been implemented and licensed as Open Source and have been validated through their deployment in a virtualized 5G environment that includes a 5G core network.

INDEX TERMS 5G networks, traffic generation, network functional testing, network functions virtualization.

I. INTRODUCTION

The progressive deployment and adoption of 5G technologies have brought new heterogeneous scenarios to the telecommunications sector, such as the development of Internet of Things networks and other new services. In the last years, many efforts have been set to deploy 5G in Europe focusing on 5G non-standalone (NSA), which allows, thanks to the DSS technology, the access to a 4G core using frequencies assigned to 5G [1]. With this technology, operators like Telefónica are supporting 5G access with over 75% coverage in Spain [2], and are working, while working with technology providers in future deployment of the 5G stand-alone (SA) version [3].

However, this initial use of 5G is not exactly ‘‘real 5G’’, as the core, where major innovation is provided, is based on 4G LTE infrastructure. Users would experiment with an increase of bandwidth but major promises from 5G as the reduction of latency will not be supported yet.

The 5G core has been designed with disruptive requirements compared to its predecessors, incorporating into its requirements the concepts of separation of user traffic and signaling (Control/User Plane Separation, CUPS) supported by Software Defined Network (SDN), virtualization (Network Function Virtualization, NFV), and deployment of customized networks (Network Slicing), along with a clear commitment to the use of server-based equipment as the base

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

infrastructure providing computing power (CPU), storage capacity (Memory) and communications capacity (Network) for core Network Functions (NF) to run. This new redesign makes it possible to reduce the dependency on specific hardware from a small number of manufacturers, opening the traditional close market in the telecommunications sector to new businesses from specialized companies.

This is the trend generally known as ''network softwarization'', aimed at the provision of network functionality based on software elements running on a cloud-like infrastructure, using open interfaces. This is the case of the Core architecture defined by 3GPP, as well as the new effort by OpenRAN [4] to split radio access functionality, so part of it may be located at a server farm, connected via standardized interfaces with physical radio devices at the towers and partially on a farm server connected by using standardized interfaces, limiting the operator's dependence between manufactures.

As indicated by Enrique Blanco [5] CTIO Global of Telefónica in 2020, this new architecture will have a significant impact on the telecommunications industry in three main areas:

- New participants: telecommunications operators will be able to choose different software components for these virtualized networks, developing more flexible and secure networks since they will not have a single provider.
- Reduction of the development cycle and implementation of new functionalities: allowing advanced functionalities such as network automation, self-optimization of radio resources, coordination of radio access nodes, and exposure to third-party multi-access edge computing applications through open Application Programming Interfaces (APIs), and their integration with virtualized core and transport networks.
- Resource sharing: Improving coverage in rural areas with difficult access or low population density, in which a sharing model is more profitable, such as in dense urban places where a large number of small cells is required.

The new requirements and challenges that arise from the use of these new technologies make it necessary to design and develop tools that can be used to test the different scenarios that will emerge, in terms of performance, security, architecture, etc. The current state of the 5G technology, along with the cost of actual 5G infrastructure deployment, makes it difficult to provide efficient ways to test the 5G environment. Taking advantage of the network softwarization process, it is possible to develop testing environments that implement the main functions of the network, at a reasonable cost and with a commodity-based infrastructure. Many proposals for the development of 5G core network functions have been proposed in the last years, but it is necessary to design and develop infrastructure around the core to build a functional test environment, including ways of measuring tests, capturing traffic, and generating traffic in the 5G network.

Addressing this kind of requirement, this paper provides a system to help in the testing and evaluation processes of emerging 5G Core solutions. This environment will be based on a set of Network Functions (NFs) coming from an ecosystem of vendors, guaranteeing the response to different scenarios or load conditions, as well as the capability to exchange data traffic and test the functional operation of the 5G access network. We are focusing on the design and development of specific tools that can be used for injecting traffic into the network, considering both signaling and user traffic that would be routed through the 5G core. This approach is interesting and novel because of the lack of currently available tools designed specifically for the definition of software-based testing scenarios over 5G networks. Moreover, this type of tool might be used for other scenarios such as Cybersecurity training in Cyber Ranges. Our developments have been released as open source and can be accessed in [6].

The rest of this paper is organized as follows:

- In section II we present a study of the background 5G-enabling technology and the scenarios that arise in this new type of network.
- In section III we provide a study of the related work on the available tools for 5G environment testing, and we describe briefly the main software-based core implementations.
- In section IV we present the architecture of the proposed system and its relationships with the 5G core and the rest of 5G infrastructure, going into more detail about each one of the two tools composing the system (for signaling and user traffic).
- Section V describes the deployed virtual testing environment and how our proposal supports the injection of signaling and user traffic in the network.
- In section VI we present the validation of the proposals by carrying out experiments using the testing environment.
- Finally, in section VII we present the main conclusions and future work for this research line.

II. BACKGROUND

In this section, we focus on the current state of 5G technology in general, as the background technology that enables and justifies our proposal.

5G technology is currently under development by 3GPP, which began its specification through releases 15 and 16 and hopes to finalize it in releases 17 and 18 [7], [8], in order to meet the expectations of the development of IMT-2020 (International Mobile Telecommunications - 2020) defined by ITU-T [9]. 5G networks are presented as a mobile access network technology with the intention of integrating 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

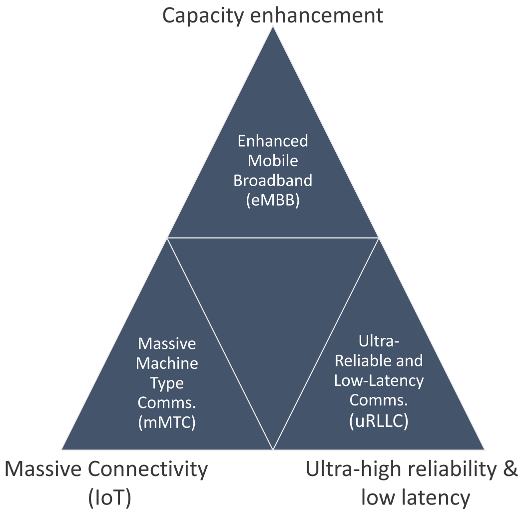


FIGURE 1. **Scenarios from IMT 2020 [9] .**

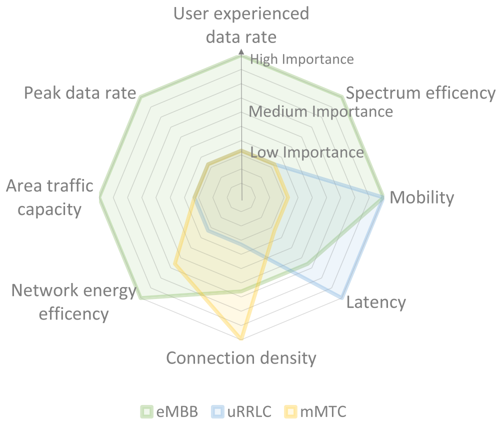


FIGURE 2. **Requirements for IMT-2020 scenarios [9] .**

networks under ''one-size-fits-all'' architecture. To support these requirements, 5G technology introduces disruptive concepts compared to previous versions such as:

- Network softwarization, a technology that aims to adapt the architecture to vertical environments.
- Software Defined Networks [10]: Data and control plane separation supporting direct programmability of network forwarding behavior.
- Network Functions Virtualization [11]: The development of network functions as software elements running on a cloud-like infrastructure.
- Network Slicing [12]: End-to-end logical networks created on an isolated physical/virtual infrastructure, with independent management and control, and deployed on demand to adapt to each scenario.

Although most of the innovation affects the Core network, in the radio part the concept of C-RAN (Cloud/Centralized-RAN) [13] introduced in 4G EPS is consolidated, considering the split of the radio subsystem into the capture/transmission of signals (antenna, Remote Radio Head (RRH)) from its processing (Baseband Unit, BBU), the latter being virtualized and instantiated depending on the traffic demand.

Based on these concepts, the IMT-2020 model identifies 3 key scenarios represented in Figure 1.

- Enhanced Mobile Broadband (eMBB): Profile designed for hitherto common services: audio, video streaming, virtual reality or immersive games derived from the increase in mobility and data transfer speed, supporting up to several Gigabits per second and high concentration of users. This will enable Fix-Wireless Access (FWA) [14], competing with the fiber-based fixed broadband service.
- Ultra-reliable and Low-Latency Communications (uRLLC): Profile designed for applications that demand high requirements regarding reliability, latency, and availability, such as Touch Internet, Intelligent Transport Systems, Vehicle to Anything Communications (V2X), Safe Transportation, Remote Medical Operations, etc. For this use cases, 5G promises delay of 1 ms.
- Massive Machine Type Communications (mMTC): Profile intended for applications for which the traffic patterns are not characterized, but based on a very high number of devices / sensors with variable delay requirements, data sensitive or not delay, and low-cost terminals with support of mechanisms of delay. In general, this group includes solutions in the field of the Internet of Things (IoT): Smart Cities, Smart Grid or Industry 4.0 are examples of these scenarios.

Figure 2 summarizes the requirements of the different scenarios based on the dependence of the demanded network parameters.

These scenarios are associated to a series of KPIs that can be seen in [15], among which some used commercially to identify the impact of the 5G network stand out, such as transmission capacities of up to 20 Gbps and latency of up to 1 ms. Starting with the IMT-2020 model, successor to the IMT-Advanced models (supported by 4G and 4G LTE networks) and IMT 2000 (supported by 3G technology), 3GPP defines the 5G network architecture. 5G technology is defined in two modes [16]:

- Non-autonomous networks or Non-standalone Networking (5G NSA), a mode intended to start the transition from 4G networks, where 4G (eNodeB) and 5G (New Radio, NR) radio access coexist, as well as both 4G (Evolved Packet Core, EPC) and 5G. This is the way in which the main operators are currently beginning to provide commercial service, mainly through the use of radio channels allocated to 5G, in the core 4G network.
- Autonomous Networks or Standalone Networking (5G SA), where both radio and core access are based solely on 5G technology and therefore can offer full-fledged 5G services, as depicted in Figure 3, where the 5G core architecture is shown [17].

This type of network, which in many cases will be deployed through software solutions, requires validation tools

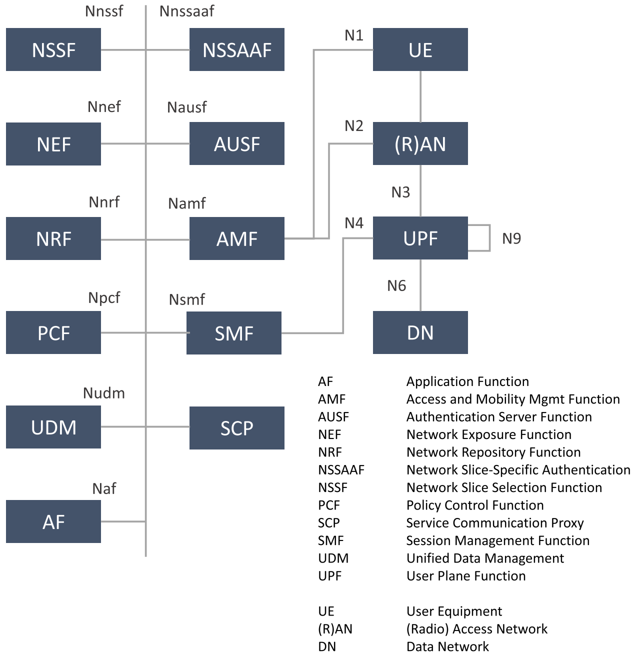


FIGURE 3. 5G core network architecture [17] .

that emulate user traffic according to different scenarios and thus allow the performance of the software modules to be evaluated.

III. RELATED WORK

This section is a review on the current testing tools available in 5G environments, and a classification of some of the available open source platforms that can be used to deploy 5G core networks, in order to generate testing scenarios.

A. 5G TESTING TOOLS

Given the relative novelty and the specific challenges surrounding 5G technology, it is crucial to count on different tools and methods to test and evaluate these networks, taking into account the difficulties related to the deployment of actual infrastructures for these purposes. In this section we review some of the main tools developed to aid in the testing of 5G networks, giving an overview of the topic and contextualizing our proposal.

According to the survey presented in [18], it is possible to classify 5G testing into four main areas:

- Technology testing: Tools and methodologies related to testing of 5G multiple access, adaptive beamforming, Full Duplex Radio technology, etc. In general, any testing of the technological aspects of the 5G network.
- Application testing: Tools and methodologies for testing the applications running through 5G networks in the different scenarios (NB-IoT (Narrowband IoT), M2M (Machine To Machine) communications, Vehicular communications, etc.).
- Architecture testing: Tools and methodologies that aim to test the 5G architecture itself, like the novel air interface, the core infrastructure, the network slicing, etc.
- Equipment testing: Tools and methodologies used to test the equipment required for the deployment of 5G networks (Antennas, terminals, base stations, etc.).

Regarding technology testing, there are studies and analyses about the multiple access techniques in 5G [19], [20], proposals about techniques and tools for the measurement of beamforming aspects on 5G [21], or the use of small cell deployments for the radio access testing [22], among other research proposals. Application testing, on the other hand, has been the goal of different proposals, like for instance, [23], where the authors propose the development of a 5G network for application testing, or [24], where they offer an integrated end-to-end platform based on vertical use cases (energy, transport, industry 4.0, etc.). Actually, these types of application-oriented end-to-end proposals can also be seen as architectural testing tools, as they allow testing the core functionalities, and the general 5G architecture. Other proposals that aim to provide tools for architecture testing are for instance [25], where they provide a way of measuring 5G architecture KPIs, or [26], where the authors propose using Digital Twins to assess 5G networks. A similar approach is currently being followed in [27]. Finally, regarding equipment testing, there are certain performance testing methodologies proposed [28], or methodologies for Over-The-Air (OTA) testing such as [29] or base stations testing such as [30].

On a more technical level, there are some tools that have been developed for the emulation of certain aspects of the 5G architecture, which can be later used for testing purposes. This is the case of the 5G core open-source software proposals that we review in the next section, which can be used to deploy 5G-capable virtual environments, or UE/gNB emulation tools, such as UERANSIM [31], that can be used for a number of different 5G scenarios as an emulator of base stations or user equipment.

Despite the wide range of projects related in one way or another to the 5G functionality testing in one of the areas that are covered by the technology (radio networks, signal beaming, network architecture, application scenarios, etc.), there are not many tools in the literature or the market that can be used to emulate scenarios where UEs send heterogeneous traffic through the network, as the ones we propose in this paper. The tool that would be closer to the one proposed in this paper is precisely UERANSIM, mentioned in the previous paragraph. UERANSIM aims to emulate both the UEs and the gNB to which they connect, implementing both sides of the gNodeB module. This allows that tool to generate user traffic into the network, but without the flexibility of allowing traffic injection from heterogeneous sources as our tools, as we allow for the injection of user traffic from any external source to the tool itself.

B. 5G CORE SOFTWARE-BASED PLATFORMS

Given the goal of this research work of providing tools for 5G testing scenarios, an implementation of the main Network Functions in a 5G core is required. There are many platforms being offered for deploying a 5G Core Network, which can be classified into commercial and open-source projects.

Nowadays it is very complex to have access to a SA 5G Core Platform. Therefore, the use of these platforms is a way of enabling deploying full testing scenarios. Most of the available platforms are still under development and are intended to offer interfaces and functions as defined by 5G standards, being developed as pure software solutions that can be run on commodity hardware. Using these platforms, it is possible:

- The development of vendor-independent software integration, openly accessible to anyone.
- Early evaluation of the interoperability and functionality developed.
- Availability of an emulated platform to facilitate the development of tools and services that can be subsequently tested in real environments.

Currently, there are different efforts in the development of 5G platforms or associated tools, differentiating open-source solutions or licensed proposals.

Among the solutions based on open source, the following stand out:

- Open Air Interface [32], led by universities and companies such as EURECOM, Fujitsu, Interdigital, Orange, Qualcomm, Nokia Bell-Labs, among others, it has been presented as a flexible platform for research, that can be used for defining realistic and controlled scalable scenarios. Recently, OpenAirInterface has teamed-up with Firecell to offer a full open-source 5G platform called Private 5G [33].
- Free5GCore [34], led by several Taiwan universities with the support of the Ministries of Education, Science and Technology, and Economic Affairs, it is a project that aims for the release of a fully open-source 5G core network, as defined in the 3GPP standards.
- Open5GS [34], a project that aims to provide users an easy way to deploy a NR/LTE network, implementing a full 5G core using C as the programming language.

Additionally, there are some proposals for the simulation of 5G networks, instead of full implementations of the network functions. This is the case of 5G k-SimNet [35], a simulation platform developed by the University of Seoul, based on the extension of the NS3 simulator to evaluate end-to-end performance in 5G networks. On the other hand, the Open Networking Foundation (ONF) has presented an open source 5G connected edge platform called Aether [36], that includes a 5G core network among other cloud-based functionalities for edge computing.

Commercial solutions are in some cases also open but require licenses to use them. Some of the most well-known proposals are:

- Open5GCore [37], led by the Fraunhofer FOKUS Institute and with extensive previous experience in platforms for SIP, IMS, EPC, etc. It aims at providing an extensive end-to-end 5G system, implementing the fundamental 5G core network functionality and providing extra functionalities (such as control and user plane separation or support for slicing). It requires licensing to be used.
- TeraVM: 5G Core Emulator [38], developed by VIAVI, it is an emulator that adds the 5G core functionality to be used with specific test mobile hardware or real User Equipment. The rationale behind this product is to provide an easy-to-deploy 5G network to test Radio Access.
- IPLOOK 5GC [39], offers a solution for 5G services based on a Service Based Architecture and a cloud-based deployment of the core functionalities.
- Landslide Core Network Testing [40], a 5G testing platform offered by Spirent, it includes test automated cases and it emulates the main network function in a 5G network.

Some companies such as Microsoft or Nokia are also offering solutions for the deployment of 5G core networks, which are not as much intended for testing but as an actual cloud-based deployment of 5G core networks:

- Nokia 5G Core [41], is the cloud-native 5G SA/NSA core solution offered by Nokia. It offers an infrastructure-agnostic design that could be deployed in any type of cloud infrastructure, adding services, programmability, and automation processes to the deployment.
- Azure Private 5G Core [42], still in a preview version, is the cloud-based 5G core offered by Microsoft. It offers the deployment of private 5G core networks on an Azure Arc–managed edge platform.
- HPE 5G Core Stack [43] is the Hewlett-Packard Enterprise alternative for a cloud-native 5G core, providing a multivendor solution for carrier environments.

For this research purpose, we have considered specifically using open-source platforms for the deployment of the testing environment, mainly focusing on simple deployment and interoperability, allowing us at the same time to look at the actual code that implements the core functionalities.

IV. PROPOSED ARCHITECTURE

Functional testing in 5G networks requires a set of functionalities that enable the deployment of a 5G core infrastructure and then the establishment of realistic scenarios to be emulated over it. Some of these functionalities, such as the virtualization of core Network Functions, or the creation of virtual connections among them and with external networks can be achieved by using existing technologies like those discussed in the state-of-the-art section of this document. Nevertheless, the emulation of user equipment and its behavior to define specific 5G traffic scenarios requires the design and development of specific tools that can interact with the virtualized infrastructure.

According to this, we have designed and developed tools for the emulation of 5G signaling traffic and tools for the injection of diverse user traffic through the 5G infrastructure,

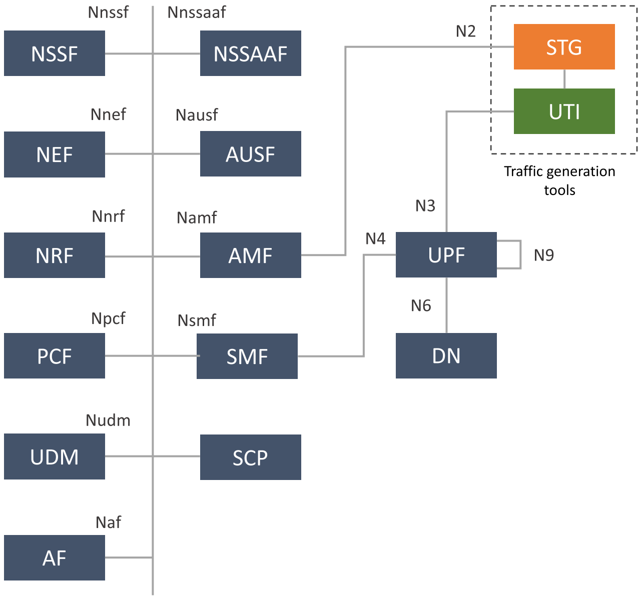


FIGURE 4. Relationship between the designed components and the 5G core network functions.

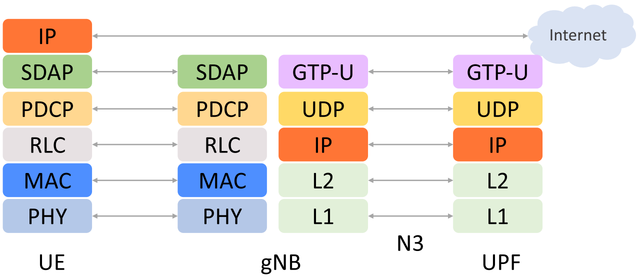


FIGURE 5. 5G protocol stack between UE, gNB and core NFs (User Plane).

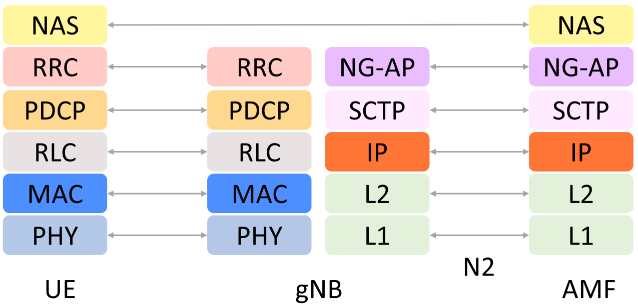


FIGURE 6. 5G protocol stack between UE, gNB and core NFs (Signaling Plane).

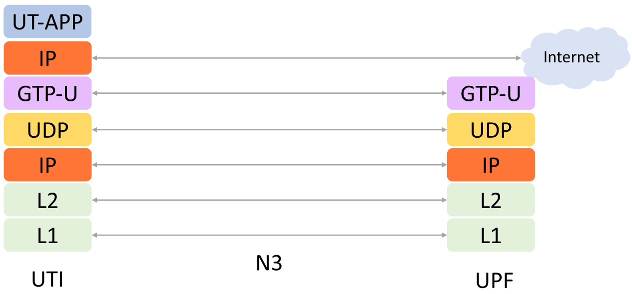


FIGURE 7. Protocol stack between the designed tools and core NFs (User Plane).

which, in conjunction with a specific infrastructure can provide a full testing platform for 5G. The goal of our proposal is, on one hand, to emulate the signals of diverse user equipment (that is, establishing sessions, performing authentication operations, etc.) without using actual hardware equipment or radio access networks, but directly communicating the emulator to the core. And, in the other hand, the rationale behind injecting data traffic is to provide a flexible tool for the generation of different types of user traffic, emulating the use of the network in different cases. The operating environment is shown in Figure 4. In this figure, we show the relationship between the designed components called Signaling Traffic Generator (STG) and User Traffic Injector (UTI), and the network functions in the 5G core. Note again that the designed solution does not require the use of a radio access network, nor a station base for the connection of emulated UEs, providing thus a completely software-based environment that can be easily virtualized.

The two main modules of our design are:

- Signaling Traffic Generator: which includes the functions necessary for the emulation of signaling operations in the Core 5G network through the N2 interface.
- User Traffic Injector: which includes the functions necessary for the emulation of user data traffic exchange with the Internet through the Core 5G by using the N3 interface.

For the design of the testing tools, it is necessary to define a terminal architecture that allows the development of a traffic emulator to evaluate the Core network functions. To do this, based on the protocol stack of the 5G data and control plane (Figures 5 and 6), the protocols implemented by the tools are shown in Figures 7 and 8. As seen in these figures, the radio access protocols with direct access to the N2 and N3 interfaces have been replaced by STG and UTI. As explained before, this would make it unnecessary to count on an actual radio network or gnB nodes for the testing environment.

The modules have been designed to implement protocols that allow signaling and user traffic communication. On top of this software, general-purpose applications can be used for the configuration of signaling traffic and injection of user traffic from external sources, depending on the scenario.

A. SIGNALING TRAFFIC GENERATOR (STG)

The Signaling Traffic Generator module is composed of the implementation of the main protocols used in the signaling between base stations and the 5G core network through the N2 interface. This allows for the evaluation of different functions related to signaling, such as authentication, session establishment, session release, etc. Along with the protocol

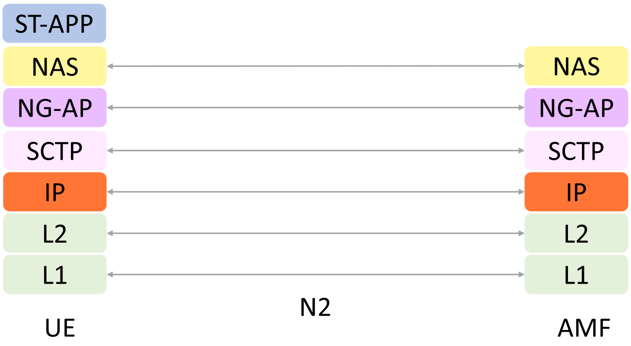


FIGURE 8. Protocol stack between the designed tools and core NFs (Signaling Plane).

TABLE 1. STG/UTI implementation use cases.

Code	Description
SUC-1	STG Registration
SUC-2	STG PDU Session Establishment
SUC-3	STG PDU Session Modification
SUC-4	STG PDU Session Release
SUC-5	STG Service Request
AUC-1	Sequential UE registrations
AUC-2	Random UE registrations
AUC-3	Simultaneous PDU sessions
AUC-4	One/Several UEs with the same traffic profile
AUC-5	Several UEs with different traffic profiles

implementation, a set of main functions have been established in this software, in order to be able to easily configure and perform the main functionalities of the most common scenarios. As a matter of fact, we have designed a set of use cases for the design of the main functionalities of this software module, and then we have built the functionalities to meet them. The use cases are based on the NAS protocol and are divided into standard use cases (SUC), based on NAS single operation defined for mobility management and session management, and advanced use cases (AUC) which is a composition of SUC considering the emulation of a set of terminals in the signaling as well as data plane. Table 1 provides the set of use cases considered.

For each use case, a detailed description is provided in a form of a table in which relevant actors, as well as procedures and dependencies, are defined. Table 2 shows the example of STG registration while Figure 9 shows the correspondence sequence diagram showing the interactions between network functions involved.

B. USER TRAFFIC INJECTOR (UTI)

The User Traffic Injector has been designed to exchange arbitrary IP datagrams through the UPF in the core, using the N3 interface. The traffic generator allows for the configuration of different types of traffic to be sent, such as Elastic (e.g., file transfer, email, instant messaging), Interactive (e.g., request/response), Streaming (e.g., audio, video), and/or Conversational (e.g., voice calls).

TABLE 2. Example of use case definition (UE Registration process from the STG).

Code	SUC-1
Name	STG Registration
Actors	STG - AMF - AUSF - UDM - PCF
Description	STG emulates the new UE entering the 5G network. It starts a NAS signaling connection by sending a NAS Registration Request message, initiating the registration procedure used in 5G services
Sequence	Registration DG-0 ->Identification DG-1 -> Authentication DG-2
Precondition	-

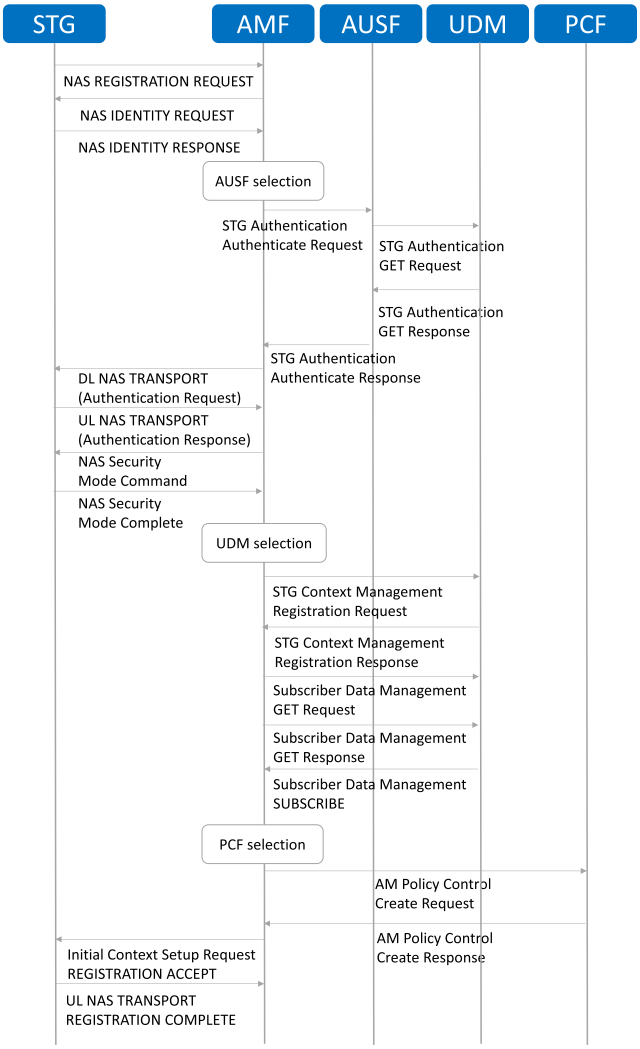


FIGURE 9. Sequence diagram for the registration of a new UE through the STG.

The required signaling for the 5G network to expect user traffic from the synthetic user equipment will be performed by the STG module, in coordination with this module, establishing a PDU session prior to any user data exchange.

The signaling process allows for the registration of new synthetic UEs in the core, by establishing a connection with the AMF. A new PDU session is also established for each

UE to synthesize, receiving specific information to use in the traffic injection performed later.

Once the sessions are established, the UTI opens a communication socket with the configured UPF. These communication channels are essentially GTP-U tunnel endpoints, and all the traffic sent to the UPF is encapsulated.

In order to allow a wider range of traffic to be sent through the 5G core, instead of generating its own traffic, the connector works as a sniffer and traffic injector at the same time. It captures all ethernet traffic received in a specific network interface and sends it through the interface connected to the UPF using the appropriate tunnel endpoint. Then, a set of application machines, are attached to the interface to generate the actual traffic.

To add link layer traffic capturing capabilities to the broker, we use the go.pkt library that can be found in [44]. This library, written in Go, uses the well-known libpcap [45] library to provide a set of functions for capturing, filtering, or injecting traffic in a network interface.

The traffic capture tool is attached to the network interface connected to each one of the Apps that will generate the UE traffic, capturing all the traffic that they generate. For each packet captured in that interface, the UTI removes the incoming Ethernet header, and encapsulates the IP datagram into a GTP-U tunnel, adding the corresponding GTP header. The tunnel endpoint (Tunnel Endpoint ID, TEID) to use in the header depends on the PDU session established for each UE. As the IP addresses assigned to each UE have also been established during the PDU session establishment, the UTI looks up the source IP address in the incoming packet, mapping it to the corresponding TEID. This allows for the simulation of an arbitrary number of UEs by connecting new Apps to the UTI interface and generating new PDU sessions. The encapsulated traffic is sent to the corresponding UPF in the core, which will oversee decapsulating and forwarding it to the data network.

Accordingly, the UTI performs a similar function for the traffic that comes from the data network to the synthetic UEs (for instance, responses from the requests made from the applications to a given server). In this case, the traffic received from the UPF is decapsulated, removing the GTP-U header and the rest of the layers from the UPF connection. Then, a new Ethernet header is set, so the IP datagram can reach its destination in the corresponding App. To generate this header, the UTI uses the destination MAC address found in the system ARP table for the destination IP address found in the incoming packet. Once the Ethernet frame is completed, it is sent to its destination through the interface connected to the applications.

V. TESTING ENVIRONMENT

A prototype implementation of the proposal has been developed, using the Go language for programming both the STG and the UTI modules. The developed software has been released using an open-source license and can be found in [6]. The implemented tools have been integrated into a testing scenario, based on a virtualized environment where we have deployed the implementation of a 5G core. As a 5G core, we have selected Free5GC [34] due to its maturity level and its open code philosophy.

Moreover, we have used part of the implementation of the protocols from the Free5GC project in our prototype, in order to avoid a reimplementation of message structures and exchange mechanisms, building the software modules above the protocol implementation, especially in the case of the STG module.

The tools have been developed considering the virtualized environment which will be deployed along with the rest of the 5G infrastructure. We have used an OpenStack environment to deploy every 5G network function along with the tools. Each network function has been deployed as a separate Docker container, connected internally by creating virtual internal links between the NFs. The containers run inside an OpenStack instance, and each module can connect to the required NFs (AMF and UPF) through the instance's network interfaces. The modules have also been deployed as a different OpenStack instance, using Linux as the operating system.

To complete the testing environment, we have configured the UPF network function to be able to reach the OpenStack network through one specific instance interface, and eventually, reach the Internet through the virtualized environment Internet connection. Other OpenStack instances have been added to the environment to act as servers (web servers, video streaming servers, file sharing servers) and clients (traffic generation virtual machines that will generate the actual traffic to be sent to the UPF in the 5G core). These clients can act as main traffic generators (emulating the user traffic from specific UEs) or as background traffic generators (emulating other traffic that can be found in the network, that can be generated by synthetic models or by traffic reinjection, giving more realism to the scenarios).

In the testing scenario, virtual networks are configured to be able to capture traffic at any point in the network, generating datasets for further analysis or other purposes.

Figure 10 shows the integrated testing environment, including the implemented tools and every other element in the scenario.

As the diagram shows, the environment is basically composed of the 5G Core NFs running in different docker containers communicated internally through docker virtual networks, the software connecting clients running in OpenStack instances running user applications (web browsing apps, video streaming apps, etc.) or can run a traffic generation client, which can be used for instance of background traffic generation, depending on the specific testing scenario requirements, and a set of local servers. These servers can also run server-side apps such as web servers, streaming services, or traffic generation servers. The environment is also connected to remote servers and to the Internet, so any Internet-based service can also be included in the testing

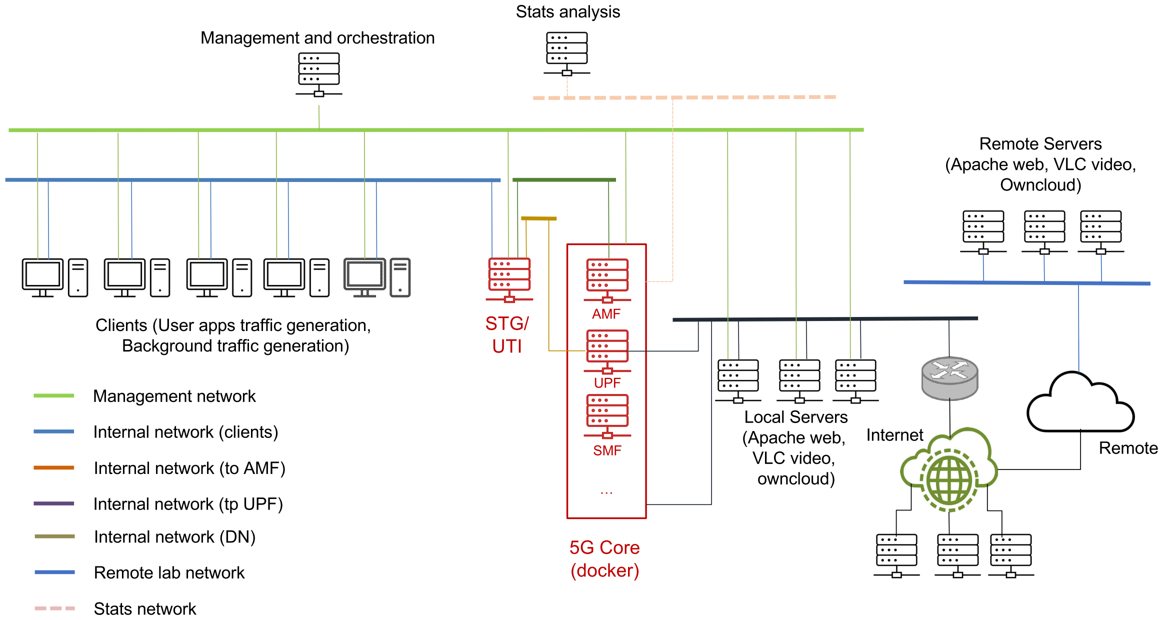


FIGURE 10. Testing scenario based on OpenStack virtual instances.

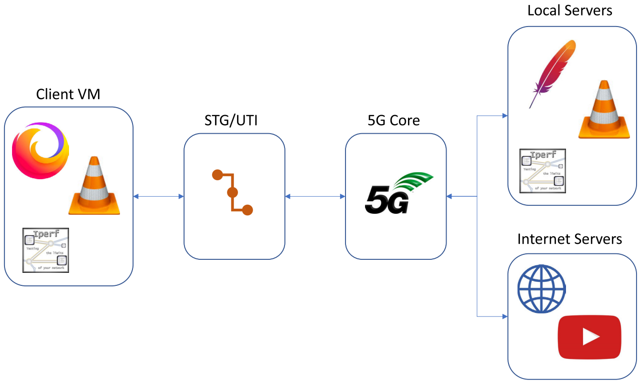


FIGURE 11. Testing applications used in the preliminary evaluation process.

scenarios. Examples of the running client and server apps are shown in Figure 11.

TABLE 3. STG/UTI implementation use cases validation results.

Code	Result	Passed
SUC-1	Registration process has been carried out from STG	✓
SUC-2	A single PDU Session has been established from STG	✓
SUC-3	A single PDU Session has been modified from STG	✓
SUC-4	A single PDU Session has been released from STG	✓
SUC-5	A single service has been requested from STG	✓
AUC-1	A sequence of UEs has been registered from STG	✓
AUC-2	A set of random UEs has been registered from STG	✓
AUC-3	More than one PDU session has been established and maintained from STG	✓
AUC-4	More than one UE is able to send traffic of a single profile*	✓
AUC-5	More than one UE is able to send traffic of different profiles*	✓

* See section V for a definition of traffic profiles.

VI. EVALUATION

Using the testing environment described in the previous section we have carried out a number of tests to validate the proposal functionality.

First, for an initial validation, we have tested individually each one of the use cases defined for the signaling functionalities, obtaining the results shown in Table 3. As the table shows, each test regarding the basic functionalities was successfully carried out.

These tests are carried out by configuring the STG tool to perform only each operation that composes each use case. The exchange of messages between the STG software and the AMF module has been monitored by capturing and analyzing the traffic using Wireshark. The results show that the message flow is correct and that the result in each case (*e.g.* A new session is established), is the one expected.

Another set of tests was carried out in the environment regarding the validation of the functionalities related to the user traffic. For this, we have selected some heterogeneous user-level applications that could be used by the clients in the virtual environment and then we have tried to use them through the 5G infrastructure (see Figure 11). Three different scenarios have been taken into account:

TABLE 4. User traffic tests carried out.

Test	Result	Passed
Web Browsing	It is possible to navigate through web pages located both in local servers and on the Internet	✓
Video Streaming	It is possible to receive and send video from local servers (VLC) and receive video streaming from Internet services (Youtube)	✓
File Downloading	It is possible to download files located both in local servers and on the Internet	✓

- Web browsing: We use the Firefox Web Browser installed in one of the client virtual machines in the testing environment to browse both a local web page served from one of the servers in the environment, and to browse Internet-located web pages, accessing them through the Internet exit in the OpenStack environment.
- Video streaming: VLC has been used to stream video between virtual machines in the testing environment, directing the video traffic through the 5G core.
- File downloading: Files located in virtual machines have been downloaded from the client's virtual machines.

These tests are essentially based on determining, from a user point of view, that "normal" user behavior in a UE can be achieved using our tools. The "normal" user behavior means that it is possible to browse web pages, download files or watch video streaming with acceptable performance. The actual performance values were not measured in these tests, as the main goal was to validate the functionality of the STG/UTI modules. The actual performance tests were measured using pre-defined traffic profiles based on actual traffic captures, using a professional traffic generation tool, as will be shown later in this section. Again, results show that it is possible to use the designed tools to generate and inject real traffic of different types into the 5G network as it is summarized in Table 4.

We have also tested the behavior of the environment when used with different heterogeneous traffic. To achieve this, we have established different types of traffic profiles based on the ITU recommendations [46], [47], and then we have tested them using a synthetic traffic generator. For these tests, we have used the IXIA BreakingPoint tool [48], which has been integrated into the virtual testing environment and configured to send traffic through the STG/UTI software.

The traffic profiles for these tests, selected in order to cover the main types of traffic that could be considered in most scenarios, are the following:

- Conversational traffic: Real-time human-based conversations, such as VoIP services, interactive games, etc. This type of traffic requires limits in delay and packet loss. A profile based on a capture from Google Hangouts conversations was used for this test.
- Interactive traffic: Used by request and response end-user services, such as chat messages, interactive web browsing, voice messaging, etc. This kind of traffic requires a low BER and a low Round Trip Delay, as it is very sensitive to transfer delays and packet loss. To test this type of traffic we have used a model based on capture for IRC chat.
- Streaming traffic: Mainly unidirectional traffic with high utilization and low time variation within a single flow. Basically, any real-time traffic where delay, jitter, and packet loss rate are not strictly limited, except for QoS requirements. In this case, traffic from Netflix streaming has been used to carry out the test.
- Elastic traffic: Used by store and forward services, like FTP file transfer, or e-mail downloading. It can be transported at random transfer rates, and it tolerates delay and jitter. However, data loss cannot be permitted, meaning that a packet recovery system is usually needed (adding more delay to transmission). It might require high bandwidth due to the size of the exchanged files or messages. In order to test this type of traffic we have used a profile based on captures from Gtalk calls.

Using these profiles we have carried out different tests, obtaining the results summarized in Table 5. To emulate each type of traffic we have used the specific profiles from real-life applications that are available within the IXIA BreakingPoint tool that have been selected for each traffic type as shown in the previous paragraphs. Although measuring the maximum performance of the tool is not the main goal of these tests, we measured the Througput (Mbps) and Frame rate (fps) for each test, in order to validate that the performance was sufficient to carry each type of traffic.

The results show that it is possible to use the proposed software to inject virtually any type of traffic in the 5G core network, with highly stable results in terms of throughput and frame rates, regardless of the actual traffic profile used in the tests.

In order to complete the evaluation process, we have defined general metrics to determine if the designed tool and the developed prototype meet the minimum required performance values to be used in virtualized realistic environments, taking into account that the testing environment has not been designed for performance tests and the available resources might not be up to real-life environment standards.

- Regarding the number of required instances for a testing environment to work properly, we have determined that at least one instance for each NF in the 5G core should be able to be deployed (although in our testing scenario we deploy NFs as containers within a single virtual instance), and a set of virtual instances that work as clients, emulating UEs within the environment. This means that the virtual environment should be able to deploy around 20 virtual machines. Optionally, a set of virtual instances containing servers or routers could be needed in the platform.
- We have tested the limit in the number of clients that can be connected concurrently to the STG/UTI deployment. We found that it is possible to connect up to 15 clients to

TABLE 5. Different traffic profiles results using IXIA BreakingPoint.

Traffic profile	Description	Test time (m)	Bytes sent (B)	Throughput (Mbps)	Frame rate (fps)
Conversational	Google Hangouts call profile test	20	311,514,537 (tx) 215,150,992 (rx)	0.6954 (tx) 0.5265 (rx)	223 (tx) 210 (rx)
Interactive	IRC chat profile test	20	504,813,840 (tx) 437,632,649 (rx)	0.5959 (tx) 0.5636 (rx)	427 (tx) 430 (rx)
Streaming	Netlix streaming movie profile test	20	463,118,979 (tx) 28,191,536 (rx)	1,88 (tx) 0.5991 (rx)	735 (tx) 642 (rx)
Elastic	GTalk file transfer profile test	20	277,928,625 (tx) 271,194,890 (rx)	0.7484 (tx) 0.7393 (rx)	330 (tx) 325 (rx)

a single STG/UTI, obtaining some performance downgrade above that number.

- Regarding the average throughput that can be obtained using the virtualized infrastructure and directing all traffic through the STG/UTI software, the tests show an average value of 0,794 Mbps, which of course will be dependent on the specific deployment resources in each case.

VII. CONCLUSION AND FUTURE WORK

As the results from the previous section show, the tools designed in this research have been validated in a realistic virtualized scenario. The validation shows that the designed tools are able to generate 5G signaling traffic that can be used to emulate the signaling process from actual UEs and gNBs in 5G networks. Additionally, the tools are able to forward any user-generated traffic from any network application that could be deployed in the environment to emulate the user traffic from one or more UEs in the network.

In other words, the developed software is able to use the signaling messages to carry out the main functions required for the emulation of UEs in the network (such as registering the UE, managing the PDU Sessions, etc.) and enables the injection of user traffic in the network, that can be obtained from heterogeneous origins (user applications, traffic generation tools, or traffic replay tools for instance).

The validation tests show that the main signaling processes (such as registration, PDU session management, etc.) have been implemented and that they can effectively communicate with the core NFs, generating new sessions for UEs and allowing further user traffic to be captured, encapsulated, and sent to the corresponding UPF in the 5G network.

The design of the STG/UTI software enables an extremely flexible way of defining different traffic profiles or mixing them to be used in testing processes of 5G networks. Its design that emulates signaling but leaves user traffic generation to a different instance or set of instances makes it easy to use or develop specific user-traffic generation applications depending on the requirements of each scenario. We have performed tests using an IXIA BreakingPoint traffic generator to inject different types of traffic in our testing scenario, obtaining similar results for every type of traffic, and validating the tool's functionality and flexibility. Furthermore, we have tested real applications such as Web browsers or video streaming tools to determine that traffic can be injected into the network.

The results also show that there is a limited throughput that we can achieve using our tool, which could lead to some limitations regarding the scenarios where this tool can be applied. This limitation has been identified as a bottleneck produced by the available resources of the core implementation that we are using for testing purposes and also by the centralized approach followed by our design in the STG/UTI tools. The first cause of this limitation could be overcome by assigning more resources to the core (which could be instantiated in different virtual instances or even in different physical servers, given the modular nature of the NFs). More UPFs could be used in the scenario to balance the traffic that passes through the core. On the other hand, the centralized design of the STG/UTI and its scalability limitations can also be avoided by using more than one instance of the STG/UTI tool deployed in the same or in another virtualized environment. Given that all communications between the STG/UTI instances, the clients emulating UE user applications, and the core NFs (AMF and UPF) are based on IP links, and that the management of the UEs is performed in the core NFs, it is possible to run more than one instance of the STG/UTI tools at the same time.

Another limitation of these tools is related to the types of user traffic it can inject into the 5G core and receive back to the emulated UEs. Although any type of application traffic can be sent through our tools, the traffic must be sent to the STG/UTI in IPv4 datagrams, as the software is currently prepared to capture IP traffic (more specifically, Ethernet frames containing IP datagrams) and uses ARP for determining the physical destination of traffic coming back from the 5G core to the client virtual machines. In any case, with some modifications, our proposed tools should be able to use the specific mechanisms used by traffic packets to be sent to the core (*e.g.* IPv6), as long as the traffic can be encapsulated in GTP-U tunnels.

The work presented here can still be improved in terms of functionality (adding new signaling use cases, for instance) and performance. For the latter, as future work, we intend to explore the possibilities of implementing the STG/UTI functionality using specialized programmable hardware.

Another line of future work is based on designing scenarios where this tool can help in terms of testing 5G networks, especially in a cybersecurity environment.

Finally, we are also planning as future research, the use of the designs presented here in the early evaluation of mobile networks beyond 5G.

REFERENCES

[1] H. Ekström. (Jul. 2019). *Non-Standalone and Standalone: Two Paths to 5G*. Accessed: Dec. 15, 2021. [Online]. Available: https://www.ericsson.com/en/blog/2019/7/standalone-and-non-standalone-5g-nr-two-5g-tracks

[2] Telefónica. (Sep. 2020). *Press Release: Telefónica Switches on 5G and 75% of the Spanish Population Will Obtain a Signal This Year*. [Online]. Available: https://www.telefonica.com/en/wp-content/uploads/sites/5/2021/03/pr-tef-5g.pdf

[3] (Jul. 2022). *Press Release: Telefónica and Ericsson Pioneer end-to-end Automated Network Slicing 5G SA*. [Online]. Available: https://www.telefonica.com/en/communication-room/telefonica-and-ericsson-pioneer-end-to-end-automated-network-slicing-5g-sa/

[4] M. Koziol, "The clash over 5G's first mile: The wireless industry is divided on open RAN's goal to make network components interoperable," *IEEE Spectr.*, vol. 58, no. 5, pp. 40–46, May 2021.

[5] E. Blanco. (Aug. 2020). *Telefonica CTIO: 5G Will Rely on Open Networks*. [Online]. Available: https://www.lightreading.com/open-ran/telefonica-ctio-5g-will-rely-on-open-networks/a/d-id/763368

[6] UPM-RSTI. *STGUTG*. Accessed: Jul. 1, 2022. [Online]. Available: https://github.com/UPMRSTI/STGUTG

[7] 3GPP. *3GPP Specification Set: 5G*. [Online]. Available: https://www.3gpp.org/dynareport/SpecList.htm?release=Rel-16&tech=4&ts=1&tr=1

[8] *3GPP: Release 17*. Accessed: Jul. 13, 2022. [Online]. Available: https://www.3gpp.org/release-17

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

[10] D. Kreutz, F. M. V. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.

[11] 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.

[12] S. Kavanagh. (Aug. 2018). *What is Network Slicing?*. [Online]. Available: https://5g.co.uk/guides/what-is-network-slicing/

[13] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann, "Cloud RAN for mobile networks—A technology overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 405–426, 1st Quart., 2014.

[14] E.-M. Oproiu, I. Gimiga, and I. Marghescu, "5G fixed wireless access-mobile operator perspective," in *Proc. Int. Conf. Commun. (COMM)*, Jun. 2018, pp. 357–360.

[15] *Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(S)*. document ITU-R M.2410-0, Nov. 2017. [Online]. Available: https://www.itu.int:443/en/publications/ITU-R/Pages/publications.aspx

[16] *The 5G Guide: A Reference of Operators*, GSMA, London, U.K., Apr. 2019.

[17] *System Architcture fo the 5G System*, document TS 23.501, V17.5.0, 3GPP, Jun. 2022. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144

[18] P. Zhang, X. Yang, J. Chen, and Y. Huang, "A survey of testing for 5G: Solutions, opportunities, and challenges," *China Commun.*, vol. 16, no. 1, pp. 69–85, Jan. 2019.

[19] Y. Wang, B. Ren, S. Sun, S. Kang, and X. Yue, "Analysis of non-orthogonal multiple access for 5G," *China Commun.*, vol. 13, no. 2, pp. 52–66, 2016.

[20] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Non-orthogonal multiple access for 5G and beyond," *Proc. IEEE*, vol. 105, no. 12, pp. 2347–2381, Dec. 2017.

[21] R. Rehammar and J. Kvarnstrand, "Using reverberation chambers to test 5G-enabled devices," in *Proc. 13th Eur. Conf. Antennas Propag. (EuCAP)*, Mar. 2019, pp. 1–5.

[22] M. Latva-aho, A. Pouttu, A. Hekkala, I. Harjula, and J. Makela, "Small cell based 5G test network (5GTN)," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Aug. 2015, pp. 231–235.

[23] E. Piri, P. Ruuska, T. Kanstren, J. Makela, J. Korva, A. Hekkala, A. Pouttu, O. Liinamaa, M. Latva-aho, K. Vierimaa, and H. Valasma, "5GTN: A test network for 5G application development and testing," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Jun. 2016, pp. 313–318.

[24] M. Gupta, R. Legouable, M. M. Rosello, M. Cecchi, J. R. Alonso, M. Lorenzo, E. Kosmatos, M. R. Boldi, and G. Carrozzo, "The 5G EVE end-to-end 5G facility for extensive trials," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2019, pp. 1–5.

[25] G. Soos, D. Ficzere, P. Varga, and Z. Szalay, "Practical 5G KPI measurement results on a non-standalone architecture," in *Proc. NOMS IEEE/IFIP Netw. Operations Manage. Symp.*, Apr. 2020, pp. 1–5.

[26] 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.

[27] A. Mozo, A. Karamchandani, M. Sanz, J. I. Moreno, and A. Pastor, "B5GEMINI: Digital twin network for 5G and beyond," in *Proc. IEEE/IFIP Network Oper. Manag. Symp.*, Apr. 2022, pp. 1–6.

[28] F. Gringoli, P. Patras, C. Donato, P. Serrano, and Y. Grunenberger, "Performance assessment of open software platforms for 5G prototyping," *IEEE Wireless Commun.*, vol. 25, no. 5, pp. 10–15, Oct. 2018.

[29] W. Fan, G. F. Pedersen, P. Kyosti, A. Hekkala, T. Jamsa, and M. Gustafsson, "Recent advances on OTA testing for 5G antenna systems in multi-probe anechoic chamber setups," in *Proc. 6th Asia–Pacific Conf. Antennas Propag. (APCAP)*, Oct. 2017, pp. 1–3.

[30] C. P. Lotback, K. Arvidsson, M. Hogberg, and M. Gustafsson, "Base station over-the-air testing in reverberation chamber," in *Proc. 11th Eur. Conf. Antennas Propag. (EUCAP)*, Mar. 2017, pp. 628–632.

[31] A. Güngör. *UERANSIM*. Jul. 2022, Accessed: Sep. 25, 2019. [Online]. Available: https://github.com/aligungr/UERANSIM

[32] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 5, pp. 33–38, 2014.

[33] Digital Journal. *Firecell and Open Air Interface Launch the First-ever Open Source 5G Platform for Enterprises and Announce a Dozen of Pilots*. [Online]. Available: https://www.digitaljournal.com/pr/firecell-and-open-air-interface-launch-the-first-ever-open-source-5g-platform-for-enterprises-and-announce-a-dozen-of-pilots

[34] *Free5GC*. Accessed: Jul. 18, 2022. [Online]. Available: https://www.free5gc.org/

[35] S. Choi, S. Kang, S. Bahk, J. Song, and T. T. Kwon, "5G K-SimNet: Network simulator for evaluating end-to-end performance of 5G cellular systems," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Oct. 2018, pp. 1–2.

[36] ONF. *AETHER*. [Online]. Available: https://opennetworking.org/aether/

[37] T. Magedanz, M. Corici, M. Smirnov, A. Weber, A. Corici, F. Schreiner, B. Bochow, D. Vingarzan, S. Covaci, J. Kocur, V. Vlad, G. Carella, and J. Mueller. (Aug. 2014). *Towards the 5G Environment: Making Research Steps with meaningufl Results, a whitepaper of the Fraunhofer FOKUS Open5Gcore Project*. [Online]. Available: https://cdn0.scrvt.com/fokus/8597fcef292698a5/d1006b4a1175/Open5GCore-WP-v1.7_web.pdf

[38] VIAVI. (2019). *VIAVI TeraVM 5G SA Core Emulator Data Sheet*. [Online]. Available: https://www.viavisolutions.com/en-us/literature/teravm-5g-sa-core-emulator-data-sheets-en.pdf

[39] IPLOOK. *Solucion De Red Central 5GC, 5G IpLook*. Accessed: Jul. 18, 2022. [Online]. Available: https://es.iplook.com/productos/5gc.html

[40] SPIRENT. (Jul. 2016). *Spirent Landslide CORE Delivers Automated Live Network Testing for UQ*. [Online]. Available: https://www.businesswire.com/news/home/20160712005075/en/Spirent-Landslide-CORE-Delivers-Automated-Live-Network-Testing-for-UQ

[41] *Nokia 5G Core (5GC)*. Accessed: Jul. 28, 2022. [Online]. Available: https://www.nokia.com/networks/5g-core/

[42] *Azure Private 5G Core—Manage 5G Networks | Microsoft Azure*. Accessed: Jul. 28, 2022. [Online]. Available: https://azure.microsoft.com/en-us/services/private-5g-core/

[43] Hewlett Packard Enterprise. (Jul. 2021). *HPE 5G Core Stack: A Preintegrated, Cloud-Native 5G Core Solution. Data Sheet*. [Online]. Available: https://www.hpe.com/psnow/doc/a50003856enw

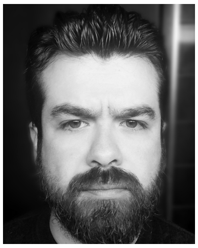
[44] A. Ghedini. (Jun. 2022). *Go.pkt*. Accessed: Jan. 31, 2015. [Online]. Available: https://github.com/ghedo/go.pkt

[45] (Jul. 2022). *LIBPCAP 1.X.Y by The Tcpdump Group*. Accessed: Apr. 14, 2013. [Online]. Available: https://github.com/the-tcpdump-group/libpcap

[46] *G.1010: End-User Multimedia QoS Categories*. Accessed: Jul. 29, 2022. [Online]. Available: https://www.itu.int/rec/T-REC-G.1010-200111-I

[47] *Y.1541: Network Performance Objectives for IP-Based Services*. Accessed: Jul. 29, 2022. [Online]. Available: https://www.itu.int/rec/T-REC-Y.1541-201112-I

[48] Keysight. *IXIA BreakingPoint*. [Online]. Available: https://www.keysight.com/us/en/products/network-security/breakingpoint.html



DIEGO RIVERA received the B.S. degree in computer science, and the M.S. and Ph.D. degrees in information technologies and communications from the Universidad de Alcalá, 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 at several nationally and internationally funded research projects. He 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 the Universidad Politécnica de Madrid, in 1996. He is currently a Full Professor with the UPM. Since 1992, he has been working at international research projects related with protocol design, 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 grid, and 5G technologies. He has published more than 100 papers in the field of advanced communications in technical books, magazines, and conferences.

MARIO SANZ RODRIGO was born in Madrid, Spain, in 1989. He received the degree in communication systems engineering from University Carlos III (UC3M), Spain, in 2017, and the M.Sc. degree in cybersecurity from the Universidad Politécnica de Madrid (UPM), Spain, in 2019. He is currently a Researcher of telematics engineering with the 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 R. LÓPEZ joined as a Senior Technology Expert at Telefonica I+D, in 2011. He is currently in charge of the technology exploration activities with the GCTIO Unit. Before joining Telefónica, 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), and the NOC of ETSI ISG NFV.

ALBERTO MOZO received the M.Sc. and Ph.D. degrees in computer science from the Universidad Politécnica de Madrid. He is currently a Professor with the Universidad Politécnica de Madrid. He has been involved in the technical leadership of several research projects funded by the European Commission related to the application of machine learning to cybersecurity, networks, and cloud management scenarios. His current research interests include network protocols and machine learning.

• • •