A NFV system to support configurable and automated multi-UAV service deployments

Borja Nogales[1], Victor Sanchez-Aguero[1,2], Ivan Vidal[1],
Francisco Valera[1], Jaime Garcia-Reinoso[1]
[1]University Carlos III of Madrid, Av. Universidad, 30, 28911, Leganés (Madrid), Spain
[2]IMDEA Networks Institute. Avda. del Mar Mediterráneo, 22, 28918, Madrid, Spain
bdorado@pa.uc3m.es,victor.sanchez@imdea.org,{ividal,fvalera,jgr}@it.uc3m.es

ABSTRACT

In this paper, we explore the strong potential of Network Function Virtualization (NFV) technologies to enable multi-mission small unmanned aircraft systems. In this context, we analyze the main challenges of using NFV technologies in this emergent field, and we present the design of an NFV system that supports the flexible, automated and cost-effective deployment of network services over small unmanned aerial vehicles. To validate our design, we implemented its most relevant components with open-source technologies, using this first prototype of the system to carry out a set of preliminary experiments that showcase its feasibility and functionality.

CCS CONCEPTS

• **Networks → Network architectures**; **Cloud computing**; *Network services*; *Mobile ad hoc networks*; Wireless access points, base stations and infrastructure;

KEYWORDS

SUAV, NFV, MANO, Virtualization

ACM Reference Format:
Borja Nogales[1], Victor Sanchez-Aguero[1, 2], Ivan Vidal[1], Francisco Valera[1], Jaime Garcia-Reinoso[1]. 2018. A NFV system to support configurable and automated multi-UAV service deployments. In *DroNet'18: 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications , June 10–15, 2018, Munich, Germany*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3213526.3213534

1 INTRODUCTION

In recent years, the continuous evolution of the Small Unmanned Aircraft System (SUAS) technologies is opening new opportunities to support civilian applications and services of great interest, not only for individuals, but also for society in general. An SUAS is typically composed by a Ground Control Station (GCS), which monitors and controls the operation of a number of Small Unmanned Aerial Vehicles (SUAVs), providing a specific service over a deployment area. SUAS services do not only include the dissemination of images and video content. On the contrary, new appealing applications and services are being actively considered by the industry and the research community, including collaborative search and rescue, remote sensing in surveillance operations, agribusiness, industrial inspection (e.g., power lines and industrial parks), monitoring (e.g., structural analysis of constructions), building aerial sensor networks to aid disaster management, or supporting backbone communications to mobile ground stations, to name a few of them.

On the other hand, we are entering in a new era where the softwarization of network functions and components is expected to play a fundamental role in the provision of upcoming telecommunication services, as enabled by the integration of information technologies and networking. Network Functions Virtualization (NFV) is an example of an outstanding technology that embraces this new softwarization paradigm, being considered as one of the fundamental enablers in the new generation of mobile networks (i.e., 5G), and under intense research and development by relevant stakeholders of the telecommunication market.

In this paper, we explore the applicability of virtualization technologies and NFV standards to enable flexible SUAS deployments, adaptable to different mission objectives. We claim that the incorporation of NFV technologies into the arena of small unmanned aircraft systems presents a novel alternative to support the flexible and cost-effective deployment of SUAS services, increasing market opportunities for commercial SUAS products, as opposed to conventional solutions, which are typically designed and manufactured to accomplish specific missions, and cannot easily and agilely be reconfigured to adapt to changing mission objectives.

With this purpose, we present the design of an NFV system capable of supporting the agile configuration and deployment of moderately complex network services over a cloud platform offered by a swarm of resource constrained SUAV equipment. In our system, SUAVs can automatically be positioned at different network locations, executing different and heterogeneous virtualized network functions (e.g., a routing function, a flight control module, a VoIP service function, etc.), which are flexibly interconnected through virtual networks that leverage wireless ad-hoc communications established among the SUAVs. The management and orchestration of these virtualized functions networks, as well as the location and trajectories of the SUAVs, is controlled from the GCS, which includes specific components to enable these functions. To validate the feasibility of our system, we implemented a first functional prototype based on open source-technologies, using this prototype implementation to carry out a set of preliminary experiments.

DroNet'18, June 10–15, 2018, Munich, Germany

ACM ISBN 978-1-4503-5839-2/18/06…$15.00
https://doi.org/10.1145/3213526.3213534

The rest of the paper is organized as follows. Section 2 presents the related work on NFV standardization, as well as on the utilization of unmanned aerial vehicles in NFV environments. Section 3 details the design of the NFV system proposed in this work. Section 4 introduces the first prototype implementation of this system, while Section 5 validates the feasibility of the proposed design. Finally, Section 6 concludes the paper and presents the short-term directions of our work.

2 RELATED WORK

As already commented, the softwarization of network functions opens new and interesting possibilities to support the fast and adaptable deployment of network services over infrastructures of SUAVs. Regarding the development of NFV technologies, ETSI is playing a major role since 2012, leading the standardization work in the field of NFV. In particular, the NFV reference architecture defined by ETSI [3] includes the following fundamental components (these components have been considered in our solution): (1) the Virtual Network Function (VNF), which is the software implementation of a physical network function (e.g., a network router); (2) the NFV Infrastructure (NFVI), composing the diversity of physical resources capable of running VNFs; and (3) the NFV Management and Orchestration (MANO) framework, responsible for coordination the operation of the NFV environment. The network services are built through the composition of multiple VNFs, and are automatically deployed over the NFVI by the MANO framework. With this purpose, this framework includes an orchestration service, a VNF management function, and a Virtualized Infrastructure Manager (VIM), being the latter in charge of controlling the compute, storage and network resources of the NFVI.

On the other hand, the utilization of NFV technologies has started to be considered by the research community, as indicated by the existing literature. Just to cite some examples, consider for instance [4] where the authors propose an architecture to enhance the situational-awareness during UAV missions, applying Software Defined Networking (SDN) and NFV technologies to deploy specific processing functionalities across the ground control stations. Another example would be [5] and [6] where the authors introduce a softwarization architecture for the collaboration among UAVs and Wireless Sensor Networks. However, a common approach in the literature is that the support of virtualization does not span the limited-capacity aerial infrastructure provided by the unmanned aerial vehicles.

3 SYSTEM DESIGN

As previously commented, the main goal of this work is to support the automated configuration and deployment of moderately complex and heterogeneous network services over a communication platform composed by SUAVs. These network services will be composed by a set of interconnected Virtualized Network Functions (VNFs), each providing the software implementation of a physical network function.

Figure 1 illustrates the approach that we are following in this work, where a mission planner could use an implementation of the MANO framework defined by ETSI [3], installed as a component of our system at the Ground Control Station, to deploy a number of VNFs over a set of SUAVs. As an example, some SUAVs might incorporate a VNF providing a Wi-Fi access point, providing network access connectivity to ground mobile stations (e.g., for example, belonging to an emergency response team that uses the SUAS deployment to handle an emergency situation); a second VNF can be deployed at every SUAV, implementing a flight control system that might be easily upgraded and changed across missions, and enabling to control the position and trajectory of the aerial vehicle from the GCS; and another VNF may provide network-layer routing functionalities at each UAV, to enable the delivery of data traffic across the multi-hop network formed by the SUAVs. Other VNFs can also be deployed over a subset of the SUAVs, to incorporate other required functionalities (e.g., a DNS service or a VoIP service function).

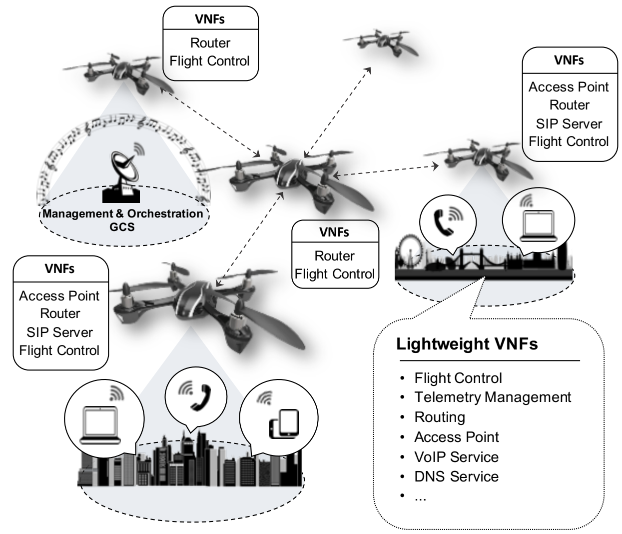

Figure 1: Proposed approach

Notwithstanding the potential benefits of using NFV in the context of SUAV platforms, it also presents a set of technical challenges that need careful consideration. On the one hand, the equipment that can be on-boarded on top of an SUAV, granting the underlying substrate to execute the VNFs, will typically be limited in terms of computing, storage and networking resources. On the other hand, the hardware platform provisioned by SUAV units should be integrated into a standard cloud computing platform as an NFV infrastructure, to be effectively used by an NFV orchestrator to deploy functional VNFs. Moreover, the control communications that are needed to manage the NFV infrastructure provided by the SUAV units must be maintained despite of their movement. Note that these requirements are uncommon in classic virtualization and cloud computing systems, where the compute nodes of these infrastructures are typically deployed as high-performance server computers interconnected through a high-speed wired network. Finally, the implementation of the VNFs must be compliant with the hardware architecture of the SUAVs.

Figure 2 illustrates an overview of our system design, which aims at supporting the management and orchestration of network services composed by lightweight VNFs, across the limited-capacity and inexpensive infrastructure that can be provided and/or transported by SUAVs. The term lightweight VNF here refers to VNFs which computational cost is not considerably high in comparison with the capacity offered by the limited infrastructure resources.

In the proposed design, there are three main components: (1) the MANO system, located at the GCS, is the system component in charge of the automated orchestration and deployment of the network services over the infrastructure provided by the SUAVs; (2) the hardware and software baseline infrastructure offered by the SUAVs, which forms the NFVI that will support the execution and interconnection of lightweight VNFs; and (3) the mission planner, also located at the GCS, which will be in charge of defining the structure and characteristics of the different network services to be deployed (i.e., their composition in terms of VNFs, the specific configuration parameters of each VNF, and the placement policies of VNFs across SUAV units), interfacing with the MANO system to request their deployment over the infrastructure provided by the SUAVs.

We want to highlight that, at this stage of our work, we are mainly concerned with the aspects related with the management and orchestration functionalities over the NFV infrastructure. For this reason, and given that the service offered by the mission planner is orthogonal to those functionalities, the paper mainly focuses on the other two components of our design. Especially on the architectural design of the infrastructure components (i.e. the SUAV devices), considering the different challenges derived from their mobility and limited-capacity.

Focusing on the design of the SUAV device, we assume that every SUAV has a wireless interface enabling the direct communication with every other SUAV that is within its radio coverage. This way, ad-hoc communications with any SUAV unit are supported, as long as that specific SUAVs is directly reachable through its wireless interface. Additionally, some SUAVs of our infrastructure will deploy a wireless access point (AP), supporting the access of mobile ground stations (i.e., users) to the aerial infrastructure (and hence, to the network services deployed on top of it).

Besides this, lightweight VNFs deployed over a particular SUAV will be interconnected through a virtual private network, which will be automatically created by the MANO system as required. A lightweight VNF providing routing functionalities, hereafter referred to as routing VNF, will also be connected to this virtual private network, providing a network relay to communicate the local VNFs of the SUAV with other VNFs hosted at other SUAV units. The routing VNF will also have an interface to the wireless AP, in case that this is deployed at the SUAV, providing network-layer connectivity to the user equipment that connect to the aerial infrastructure. Additionally, the routing VNF will provide the functionality of a DHCP server, supporting the automatic network configuration of this user equipment. On the other hand, we consider a specific type of VNF that will provide flight control functionalities. This VNF will be configured by the MANO system (i.e., particularly by the VNF manager component) at each SUAV, according to the mission planner indications, with specific information regarding the position and trajectories of the SUAV.

As a specific design criterion regarding the deployment of virtual functions, we want to highlight that given the limited-capacity hardware and software platforms that can be provided and/or transported by SUAVs, we have decided to use container virtualization, as opposed to traditional hypervisor-based virtual machines, to support the deployment of the lightweight VNFs in our design.

Finally, to support management operations towards the infrastructure of SUAVs (e.g., to start a container, to instantiate a VNF, or to terminate a VNF instance), every SUAV unit of the infrastructure must support two types of control communications: (1) communications between the SUAV and the MANO system (labeled as infrastructure management in Figure 2), to support the management of compute, storage and network resources of the SUAV; (2) communications between the VNFs and the MANO system (denoted as VNF management in Figure 2), to support the lifecycle management of the lightweight VNFs.

4 PROTYPE IMPLEMENTATION

We implemented a preliminary prototype of our design, including its most relevant components, using open source technologies. With respect to the MANO system, we used the orchestration service provided by Open Source MANO Release TWO [1], which provides the software implementation of the functionalities corresponding to the orchestration service and the VNFM manager of the ETSI NFV reference architecture. For the VIM, we used OpenStack Ocata[1]. Both the OSM stack and the VIM were deployed over a separate mini-ITX computer (Intel Core i7 2.3 GHz, 8 GB RAM, 128 GB SSD, 4 GbE ports with DPDK capabilities).

With respect to the SUAV platforms, we used a number of aerial vehicles Parrot AR.Drone 2.0, each carrying a single board computer Raspberry Pi 3 Model B (RPi). These RPis were used as the compute nodes of our design, supporting the deployment of the lightweight VNFs. Each RPi includes two Wi-Fi interfaces (an integrated interface and an external Wi-Fi USB adapter), enabling wireless ad-hoc communications with other SUAVs, as well as the deployment of a wireless access point to provide network access connectivity to mobile ground stations (in a prior research work [7], we validated the suitability of these muti-interface devices to support multimedia communications).

We incorporated the RPi boards as compute nodes of the OpenStack VIM, doing the necessary configurations to enable virtual networking through Linux bridges[2]. With respect to the virtual networks that enable the communication between SUAVs, we used Virtual eXtensible Local Area Networks (VXLAN) [2]. VXLANs present a feasible solution to interconnect the routing VNFs of different SUAVs, as: (1) they can be dynamically created by the VIM, as instructed by the OSM stack, to interconnect two routing VNFs hosted by different SUAVs; (2) VXLAN traffic can be sent over the Wi-Fi interface in ad-hoc mode that is available at every SUAV (directly sending traffic from the routing VNF, deployed over a virtualization container, through a Wi-Fi Ad-hoc network is challenging, as this is not currently supported by the Linux kernel of the RPis); and (3) the utilization of VXLANs does not require the configuration of network routes at the RPi boards (IP packets are

[1]OpenStack Ocata (last access: March 2017) https://releases.openstack.org/ocata/
[2]Bridging and firewalling (last access: March 2018): https://wiki.linuxfoundation.org/networking/bridge

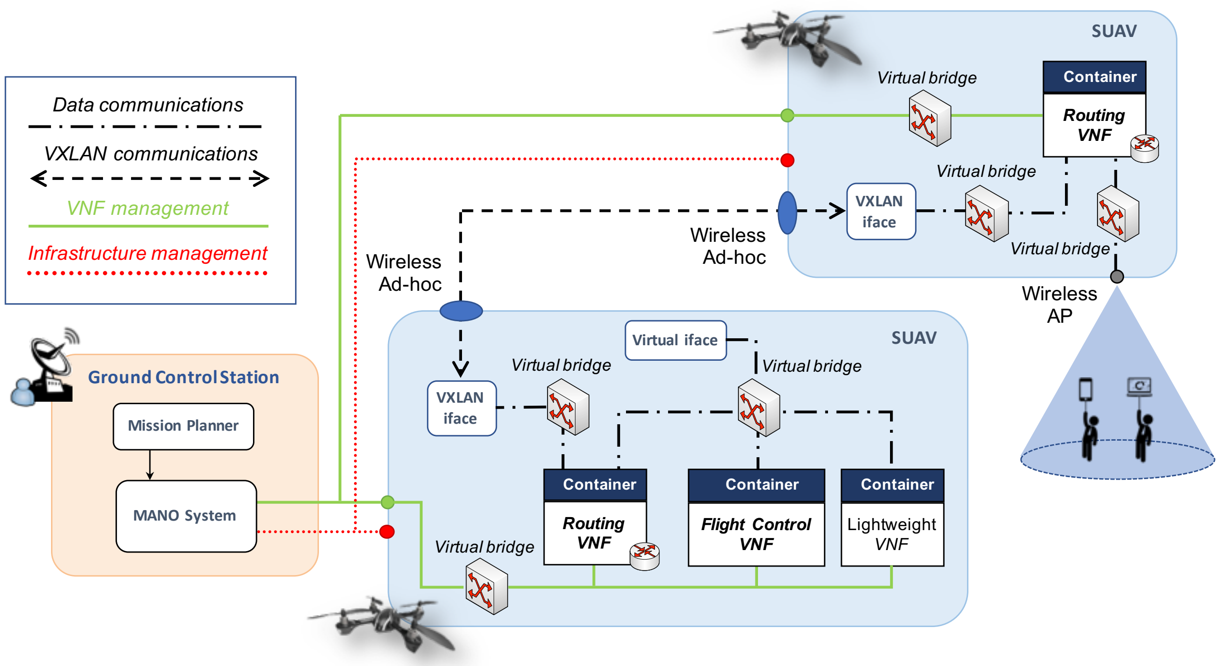

Figure 2: Overview of the platform design

directly exchanged between SUAVs using the IP addresses of the source and destination Wi-Fi interfaces).

To advance on the experimentation process, and obtain preliminary performance results regarding the feasibility of our solution (see Section 5), the control communications of our first prototype, mentioned in the previous section, have been implemented using the Ethernet interface of the RPi boards. Our short-term work includes the utilization of wireless network technologies to support these control communications. In any case, we want to highlight that data communications are already supported by our prototype through the Wi-Fi cards of the SUAVs,

Finally, to support experimentation activities, we carried out the implementation of the routing VNF. For this purpose, we used Linux containers (LXC[3]) and Ubuntu Server as the operating system, impleneting the VNF as a Linux router. The requirements of our routing VNF in terms of resources are 1 Virtual CPU, 128 MB RAM and 4 GB of storage.

5 PRACTICAL VALIDATION

We used our implementation to carry out a number of experiments, aiming at validating the feasibility of the proposed system design to support the deployment of network services over the infrastructure of SUAVs. In the following, we present the preliminary results achieved in these experiments, where flight operations have not been considered as this is not still supported by our prototype implementation.

[3]Infrastructure for container projects (last access: March 2018): https://linuxcontainers.org

In first place, we explored the performance offered by our solution to instantiate VNFs and deploy network services. Accordingly, in a first experiment we studied how the number of VNFs composing a network service affects the total time needed to successfully complete the instantiation of the VNFs and therefore, the network service deployment. With this purpose, we used our MANO platform to instantiate a network service over the SUAV platform, varying the number of its constituent VNFs from one to four. In all the cases, the VNFs used to build the network service were instances of our routing VNF. In addition, we deleted any existing network service before each deployment, so as to guarantee that existing network services did not affect the instantiation times of the experiment. For each case (one, two, three and four VNFs), we repeated the deployment 30 times. Figure 3.a illustrates the results of the experiment, where we observe that the time required to deploy a network service linearly increases with the number of its constituent VNFs (the time required to instantiate a network service with an additional VNF increases by less than 20 s). Considering the key performance indicators established by the 5G Infrastructure Public Private Partnership [9], that indicate the average service creation time should not exceed 90 minutes, we observe from the previous results that our solution has the potential to agilely build moderately complex network services with appropriate service creation times.

In a second experiment, we evaluated whether the instantiation delay of a network service is impacted by other network services already running at the platform. In this case, we considered a network service with a single VNF, and we used the MANO platform to consecutively deploy four instances of the service. As opposed to

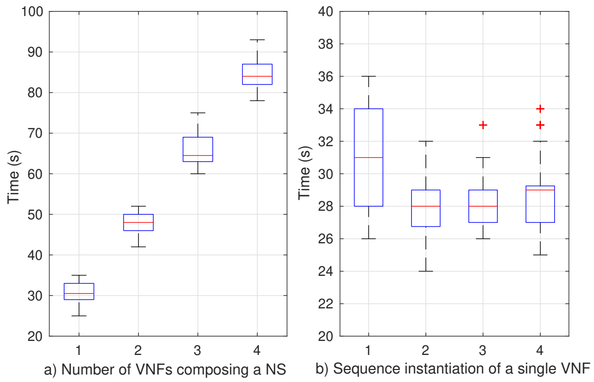

Figure 3: Comparison of deployment times

the previous experiment, network services were not deleted after their deployment. For each of the four deployments, we measured the time needed to complete them, and we repeated the experiment 30 times (deleting the four network services before each repetition). Based on the obtained results, which are represented in Figure 3.b, we conclude that having several network services running at our platform does not affect the deployment times of subsequent network services. In addition, we observe that the first instantiation takes longer, with an average time higher than 30 s.

To get an insight into the performance that could be achieved with deployments involving multiple SUAV units, we built a network service with two routing VNFs, interconnected through a VXLAN over the Wi-Fi adhoc network. We used the placement policies of the OSM stack to indicate that both VNFs should be located at different SUAV units. After deploying the network service, we connected a laptop to the wireless AP provided by one of the SUAVs, being this end-user equipment automatically configured with an IP address through the DHCP server provided by the routing VNF of the SUAV. Analogously, a second laptop was connected to the wireless AP of the second SUAV unit. This way, network connectivity was enabled between both laptops, through the routing VNFs running at the SUAVs. With this setup, we used the *Iperf* tool[4] to obtain a first estimation of the performance of the data path between the end-user equipment. The results indicated an average throughput of 5.86 Mb/s and an average jitter of 17.01 ms, which are reasonable performance figures for the multimedia services under consideration in SUAV deployments[5]. We want to highlight that all the devices were placed at a close distance, within a laboratory environment, for all the experiments covered in this section.

Analyzing the previous results, we decided to corroborate the capacity of our network service, conformed by virtualized functions over the infrastructure of SUAVs, to support the delivery of real-time interactive data. In particular, we considered the data traffic

[4] *Iperf*, the ultimate speed test tool for TCP, UDP and SCTP (last access: march 2018): https://iperf.fr

[5] Despite of these being considered as appropriate values, we want to note that in our experiments we used the same wireless channel for the APs and the ad-hoc network. This affected the throughput metrics, which could be enhanced with a careful selection of wireless channels in a real deployment

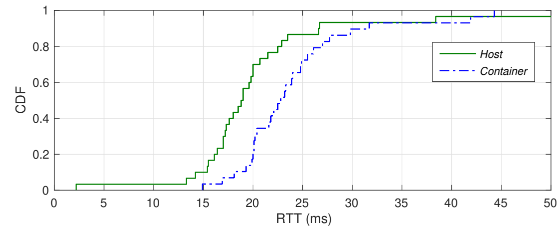

Figure 4: Round Trip Time

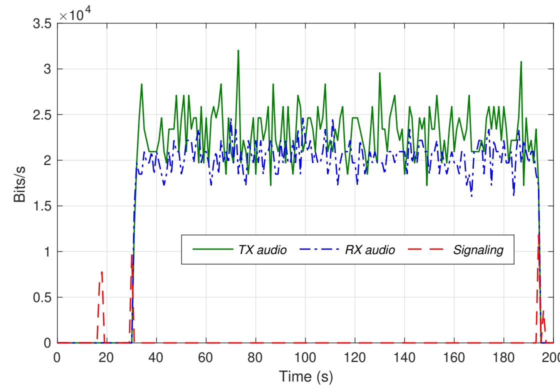

Figure 5: Data exchanged during the VoIP call

corresponding to a VoIP call, which imposes very stringent requirements in terms of end-to-end delay and network jitter. As a prior step to establish a VoIP call over the SUAV infrastructure, we executed an experiment to estimate the end-to-end delay provided by our network service. With this purpose, we took 30 measurements of the round-trip time (RTT) between the laptops, using the Ping command. Figure 4 shows the Cumulative Distribution Function of the measured RTT values. In this figure, we also show the distribution function corresponding to a network service that interconnects both laptops with two SUAVs and no virtual functions. As it can be observed, both cumulative distribution functions are similar, with 90% of the RTT measurements equal or lower than 35 ms in both cases. These results suggest that the utilization of virtualized network functions does not significantly impact the performance in terms of end-to-end delay, being the delay figures in this case appropriate for a successful VoIP call (delays lower than 150 ms [8] are indistinguishable in an interactive voice communication).

Finally, we executed a real VoIP call, using two wireless VoIP phones ZyXEL Prestige 2000W. We connected each phone to one of the APs provided by our network service, and we established an audio call across the routing VNFs of the SUAVs.

Figure 5 shows the traffic exchanged between both terminals, including the signaling messages to establish the call and the voice packets transmitted and received by one of the wireless phones. The call proceeded as expected, with no glitches and appropriate sound quality. Finally, figure 6 represents the network jitter in the forward and reverse directions, with an average value lower that 20 ms, corroborating our estimations with the *Iperf* tool.

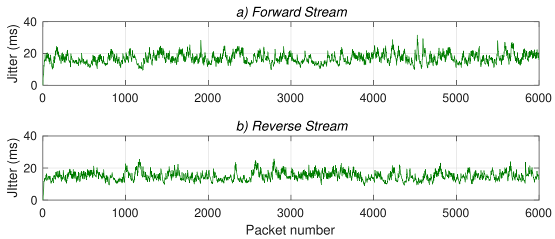

Figure 6: Jitter during the VoIP call

6 CONCLUSION

In this paper, we presented a work in progress on an NFV system to support the agile configuration and deployment of network services over an infrastructure of SUAVs. Our experimental results suggest that the proposed system can effectively support the automated deployment of network services, composed by lightweight VNFs, over the limited-capacity compute and network resources provided by SUAV platforms. Overall, our implementation presents a cost-effective solution in terms of service creation times, despite of the linear increase of the instantiation delays with the number of virtual network functions. An exploratory performance evaluation indicates that the utilization of virtualization technologies does not prevent the appropriate execution of demanding real-time services, which can be supported by the underlying hardware platforms, such as a VoIP call. In the short-term, our work includes the utilization of wireless network technologies to support the control communications of our implementation, the development of new lightweight VNFs to enable testing with moderately complex network services, and additional practical experimentation to precisely determine the tradeoffs of using virtualization technologies in unmanned aircraft system.

7 ACKNOWLEDGEMENT

This article has been partially supported by the European H2020 5GinFIRE project (grant agreement 732497), and by the 5GCity project (TEC2016-76795- C6-3-R) funded by the Spanish Ministry of Economy and Competitiveness.

REFERENCES

[1] Adrian Hoban et al., "OSM Release TWO, A Technical Overview", ETSI OSM Community White Paper, April 2017.
[2] M. Mahalingam et al., "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, Internet Engineering Task Force (IETF), August 2014
[3] ETSI GS NFV 002 V1.2.1, "Network Functions Virtualization (NFV); Architectural Framework", version 1.2.1, December 2014.
[4] Kyle J.S. White et al., "A programmable SDN+NFV-based architecture for uav telemetry monitoring" Consumer Communications & Networking Conference (CCNC), 2017 14th IEEE Annual. IEEE, 2017.
[5] S.Mahmoud et al., "A Softwarization Architecture for UAVs and WSNs as Part of the Cloud Environment", Cloud Engineering Workshop (IC2EW), 2016 IEEE International Conference on. IEEE, 2016.
[6] S.Mahmoud et al., "UAV and WSN softwarization and collaboration using cloud computing", Smart Cloud Networks & Systems (SCNS). IEEE, 2016.
[7] V. Sanchez-Aguero et al, "Deploying multimedia services over networks of interconnected MAVs", Internet Technology Letters, 2017.
[8] ITU-T Recommendation G.114, "General Recommendations on the transmission quality for an entire international telephone connection; One-way transmission time", May 2003
[9] Public Private Partnership in Horizon 2020, "Creating a Smart Ubiquitous Network for the Future Internet", Advanced 5G Network Infrastructure for the Future Internet, November 2013.