

# Dynamic SDN Controller Placement in a LEO Constellation Satellite Network

Arled Papa\*, Tomaso de Cola<sup>†</sup>, Petra Vizarreta\*, Mu He\*, Carmen Mas Machuca\*, Wolfgang Kellerer\*

\*Chair of Communication Networks, Technical University of Munich

<sup>†</sup>DLR- German Aerospace Center, Institute for Communications and Navigation

\*arled.papa@tum.de, <sup>†</sup>tomaso.decola@dlr.de, \*petra.vizarreta@lkn.ei.tum.de, \*{mu.he, cmas, wolfgang.kellerer}@tum.de

**Abstract**—Software Defined Networking (SDN) has been identified as a potential approach to achieve a more flexible control and management of the traditional satellite systems and enhance the opportunities for future services including the possibility of a hybrid satellite/terrestrial network. Given the renewed interest towards Low-Earth-Orbit (LEO) constellations, an interesting research topic is the design of a suitable network management model taking into account user specific metrics. In this paper, we address this issue while investigating the use-case scenario of an SDN-enabled satellite space segment. A Dynamic Controller Placement Problem (DCPP) is considered for a LEO constellation where the traffic demands change dynamically based on users' geographical position and time zone. To this end, we develop a mathematical model and formulate it as an Integer Linear Programming (ILP) guaranteeing an optimal controller placement and satellite-to-controller assignment minimizing the average flow setup time with respect to the traffic dynamics. We show results for the DCPP regarding the average flow setup time. Furthermore, a comparison with respect to the static approach is investigated and the proposed SDN-enabled LEO constellation architecture is compared with alternative architectures proposed in the state of the art.

**Keywords**—SDN, DCPP, LEO constellation.

## I. INTRODUCTION

The tremendous growth in user traffic demands and the requirements for ubiquity increases the trend of satellite-terrestrial network integration. The integration of the two systems has been observed as a promising approach to further improve the delivery of communication services [1].

Software Defined Networking (SDN) has been identified as the main approach to achieve this unification. SDN offers flexible control of the network and increase in the network programmability. The introduction of use-cases of SDN and Network Function Virtualization (NFV) technologies in the satellite communications are provided in [2] and [3]. These papers quantify the benefits of SDN/NFV, however most of the attention is placed on the satellite ground segment, whereas a possible SDN controller placement on the satellite space segment has not been investigated yet.

Satellite Communication (SatCom) systems have been identified as potential solutions for backhauling and supporting terrestrial networks including coverage in rural areas also offloading and balancing the traffic in dense populated areas. Low Earth Orbit (LEO) mega-constellation systems, which involve hundreds of satellites, are gaining increased attention in achieving the integration with terrestrial technologies [4].

Given the renewed interest towards LEO mega-constellations, an interesting research topic is the design of a suitable network management model using SDN. Leveraging the benefits coming from both systems, this model has the potential to achieve a more flexible control and management of the traditional satellite systems, as well as to enable a future hybrid satellite/terrestrial network.

The implementation of SDN relies on a logically centralized control plane which contains a global view of the whole network. The controller is responsible for updating the forwarding rules of the network elements' in the data plane. The time required for a rule to be installed is referred to as the flow setup time. In a large-scale network such as a LEO constellation, due to the limited resources that a controller possesses, it might result that one controller is not suitable to handle the requests originating from the data plane. Furthermore, due to the large distances between the satellites and the controller, the acceptable control plane latency is not guaranteed.

Consequently, the need of a distributed control plane becomes mandatory. References [5] and [6] distribute the control plane logic over several controllers in order to achieve better manageability of the network traffic. Nonetheless, adapting the number of controllers and their locations depending on the traffic load is an arising issue when such an architecture is implemented. The new problem is known as the Dynamic Controller Placement Problem (DCPP).

The effect of flow dynamics on the flow setup time has been studied by [7] and [8]. However, the aforementioned works do not consider the topology dynamics, where due to the satellite movement the communication links to the ground segment change over time. To the best of our knowledge, we are the first to investigate the SDN controller placement in a LEO constellation and provide results regarding the average flow setup time, considering both spatial and temporal traffic variations. This paper focuses on the investigation of the impact of the traffic variations in the average flow setup time and makes the following contributions: 1) proposes an SDN-enabled LEO constellation architecture and investigates the DCPP in a dynamic satellite topology; 2) introduces a mathematical model for the traffic exchange between the data and the control plane based on a realistic flow profile which considers both spatial and temporal user traffic requirements; 3) formulates the problem as an Integer Linear Programming

(ILP) and provides a metric to evaluate the system and compare it with other approaches.

The remainder of the paper is organized as follows. In section II, the SDN controller placement problem related work both for terrestrial and satellite networks is summarized. In section III, the SDN-enabled LEO constellation architecture is introduced together with the traffic modeling. Furthermore, the mathematical model for calculating the traffic exchange between the architecture planes is described. Finally this section is concluded with the optimization problem formulation which minimizes the average flow setup time. An analysis of the system performance with respect to this performance metric will be illustrated in section IV. Moreover, evaluation results and comparisons with alternative architectures will be presented. The paper is concluded with section V, where conclusions are drawn and an outlook of the future work is presented.

## II. RELATED WORK

### A. SDN controller placement terrestrial networks

Heller et al. in [9] was the first work to investigate the controller placement problem. The problem is formulated as a general facility location problem. An analysis of the effect of controller locations on the average as well as worst switch-to-controller latency was conducted. Further investigation was done for the controller placement problem considering different metrics such as inter-controller latency and load balance between the controllers in [10]. The authors in [11] study the controller placement regarding the reliability of the system. However, none of these papers take into consideration the time variations of the flows, by keeping the controller placement and switch-to-controller assignment fixed. Such a phenomenon can cause degradation to the system and can influence Quality of Service (QoS).

In order to be able to react to traffic fluctuations, a dynamic SDN controller placement has to be developed. Bari et al. in [8] provided an ILP and proposed two heuristic algorithms to dynamically adapt the number of controllers with changing flow dynamics. The objective was to minimize the average flow setup time and communication overhead. In [7], the controller placement is analyzed for a dynamic flow scenario. The authors in this paper aim to minimize the average flow setup time and they formulate the problem as an ILP. Two scenarios, one for optimizing the controller location and the other for switch-to-controller assignment optimization were shown and compared to the joint optimization. However, none of the aforementioned approaches takes into consideration a dynamic topology. Due to the satellites movement, the communication links to the satellite ground segments change over time. Consequently, an accurate traffic matrix referred to as flow profile, should account for traffic variations depending both on time and users' geographical position.

### B. SDN controller placement satellite networks

OpenSAN presented in [12] is the first architecture to introduce the SDN controller concept on the satellite space

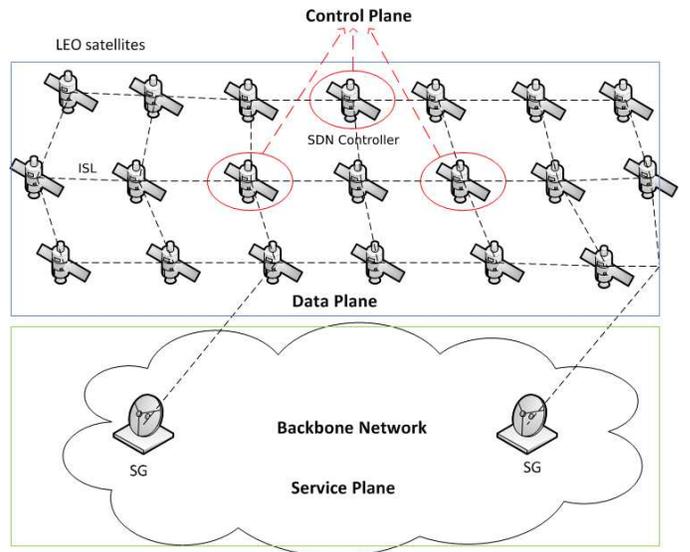


Fig. 1. Proposed SDN-enabled LEO constellation architecture.

segment. In the proposed architecture, GEO satellites are used to serve as SDN controllers. However, only the architectural concept and benefits are presented by the authors. In [13], the authors used and based their work on the architecture introduced by OpenSAN. Additionally they introduced the concept of NFV into the satellite ground segments. By exploiting the advantages of NFV, they are the first to provide results regarding the site-diversity scenario shown in [2]. However, they only consider a prototype network and the user traffic demands are not taken into account. Moreover, the SDN controller placement is not investigated. In [14], the authors assume the implementation of an SDN-enabled LEO constellation to show improvements in the throughput when a multi-path Transmission Control Protocol (TCP) approach is realized. Nonetheless this paper mostly focuses on the throughput benefits rather than the controller placement problem.

## III. SYSTEM MODEL AND PROBLEM FORMULATION

### A. SDN-enabled LEO constellation architecture

The proposed architecture of the SDN-enabled LEO constellation is illustrated in Fig. 1. As denoted, the architecture consists of three layers. The data plane layer consists of the LEO satellites of the IRIDIUM constellation. Despite being a rather old LEO constellation, we decided to consider IRIDIUM because it provides an acceptable initial and general network model for our simulations. 66 satellites are distributed into 6 orbital planes according to information provided from [15]. The control plane layer contains several LEO satellites depending on the traffic demand. Furthermore, there are in total 7 Satellite Gateways (SG) placed on the ground and serve as entry points to the backbone network. Their position is fixed and is defined from the existing IRIDIUM system.

We define the topology as a graph  $G = (V, E)$ , with vertices  $V$  representing the network nodes, in our case

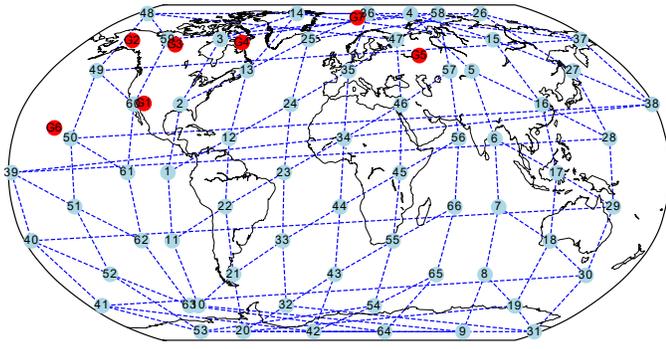


Fig. 2. IRIDIUM network illustrated in for one network snapshot. The satellite nodes are shown with blue circles, whereas the SG are denoted with red circles.

satellites and gateways, whereas edges  $E$  representing the communication links in the network. Each satellite of the constellation contains 2 stable inter-satellite links (ISLs) with satellites on the same orbit and 2 stable ISLs with the satellites on the neighbor orbits. The communication links between the satellites and the SG are not stable and hereby change over time. In order to capture this, we introduce the notion of snapshots. A snapshot captures one network topology at a given time. We compute the set of snapshots  $R$  every one hour. This set contains all the topology changes  $r$  that are introduced in the network due to links being up or down. Consequently, for each hour the set of snapshots  $R$  is identified and therefore  $|R|$  network topologies are created. One snapshot of the network topology is shown in Fig. 2.

Each satellite  $s \in S \subseteq V$  in the network hosts an SDN switch which can be only assigned to a controller at a time. Considering a distributed control plane,  $K$  controllers are allowed to be placed in the network. Each satellite in the network is capable of hosting the control unit. The selection of the satellites' in the control plane is based on the dynamic controller placement. The communication between the satellites of the data plane and control plane is realized using the existing network links (in-band control). The shortest path algorithm is applied for the routing and forwarding latency between two nodes  $s$  and  $k$  at each network snapshot  $r \in R$ , which is denoted to as  $d_{r,s,k}$ . The satellites part of the control plane serve both as controllers and normal network switches. They manage, control and update the forwarding rules of the flow tables of the satellites of the data plane. On the other hand, the satellites being part of the data plane are only responsible for forwarding packets based on rules defined from the corresponding controllers.

### B. Data plane traffic modeling

The modeling of the data plane traffic considered in this paper is based on [16]. The earth is divided into  $15^\circ \times 15^\circ$  geographical zones. In total there are 288 such zones. Each geographical zone  $i$  corresponds to a square region, whose requirement density is based on the forecast of voice traffic for the year 2005 and introduces the user geographical position

traffic demand  $w_i$ . Additionally, a user hourly activity is introduced to obtain the temporal user demand depicted as  $\tau_i$ . A mapping between the square regions and the satellites which cover the specific areas is realized by checking the satellite flying over that region. In our work, we use the notion of region square requirement density with the assumption that the IP traffic density requirement of each region is proportional to the number of active users while adapting it to today's internet traffic demands. We use STK [17] to quantify the position of the regions. Due to STK's special feature *grid coverage*, the number of regions is reduced to 192. Therefore, a new model is created while mapping 288 regions of the theoretical model to 192 regions, based on the regions' coordinates. Furthermore, differently from [16], we provide a daily user activity model based on [18]. Each region communicates with the SG using the LEO satellites. At any time any region is mapped to the satellite flying over it. Therefore, the flow profile contains satellites as sources and SG as destinations.

The IP traffic rate generating from each region  $i$  is calculated based on (1). For each region  $i$ , the IP traffic rate depends on the ratio of region's traffic requirement, compared to the overall regions' traffic requirements multiplied by the total offered traffic  $T_A$  and temporal traffic requirement of the region. We assume that the traffic is uniformly distributed among the 7 SG. Thus the IP traffic rate of a region  $i$  to a specific destination is  $\lambda_i/7$  and is expressed in *Mbps*.

$$\lambda_i = \frac{w_i}{\sum_{k=1}^{192} w_k} \cdot T_A \cdot \tau_i. \quad (1)$$

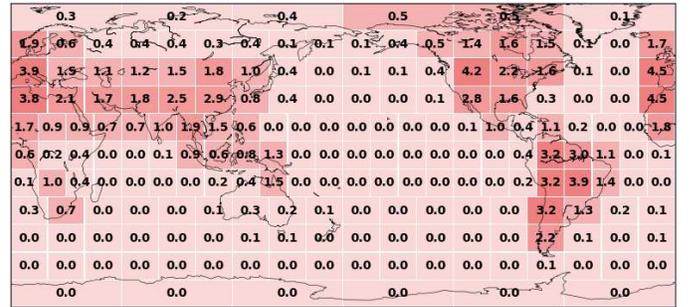


Fig. 3. Traffic distribution of regions in terms of new flows per second for  $\tau_i$  10 AM GMT.

Finally, the IP traffic is converted into flows per second by dividing the data rate by the average flow size. The average flow size is calculated based on the flow characteristics for the most relevant service types provided by the system. However, considering an SDN architecture, only a portion of the flows that exist in the system, named as new flows per second could invoke the controller. Therefore, a variable  $\eta$  is set to 10% to quantify the ratio of new flows. Eventually, the traffic model for each region  $i$  is expressed in number of new flows per second and is used to compose the flow profile  $F$ . The corresponding model is illustrated in Fig. 3.

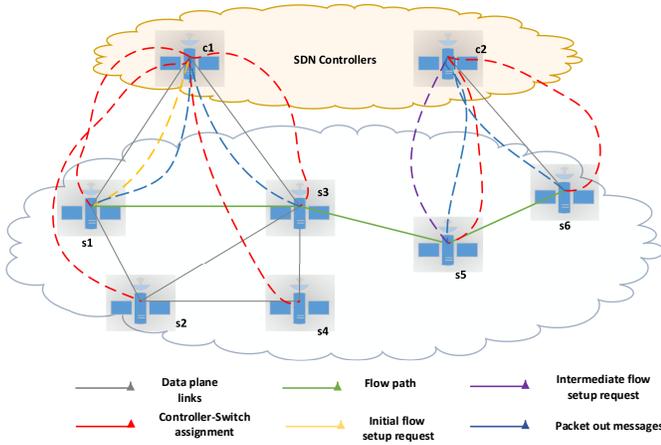


Fig. 4. Illustration of the flow setup process upon a new flow arrival for a multi-domain system.

### C. Control plane traffic modeling

Due to the separation of the data and control plane implied in the SDN architecture, the implementation of a model to distinguish the traffic among the two layers becomes mandatory. The protocol used for the communication between the aforementioned planes is OpenFlow [19]. There are several message types provided by OpenFlow, however in this paper we only focus on Packet-In messages which have a direct impact on the flow setup time.

Considering a multi-domain system as shown in Fig. 4, the flow setup time is defined as follows. Upon a new flow arrival at satellite  $s_1$  of the data plane which does not contain forwarding information about the specific flow, a request to the corresponding controller  $c_1$  is initiated. This request is called initial flow setup time request. A Packet-In message is therefore sent by  $s_1$  to  $c_1$ . After the flow path is calculated by  $c_1$ , flow rules are properly decided and sent to every involved satellite in the flow path part of controller's domain, in this case  $s_1$  and  $s_3$ .

Similar to our case, for a specific flow, the flow path might contain data plane satellites part of a different domain rather than the one where the flow was initiated. In such a case, an intermediate flow setup request is initiated. Satellite  $s_5$  initiates the intermediate flow setup request by sending a Packet-In message to its controller  $c_2$ . Similarly, once the flow rules are obtained,  $c_2$  updates the flow tables of the involved satellites in the flow path,  $s_5$  and  $s_6$ . Once the flow reaches the destination the whole process is finished. Therefore, the flow setup time is calculated as the difference between the time satellite  $s_1$  receives the first packet of the flow and the time satellite  $s_6$  successfully delivers it to the destination.

According to [7], the flow setup time is the sum of 2 times the flow setup request and intermediate flow setup requests if any, and forwarding latency between the source and the destination of the flow. Consequently, according to (2), the first term of the equation provides the forwarding latency between the source and destination, whereas the second term

adds the latency acquired due to flow setup and intermediate flow setup requests for each flow  $f$  of the flow profile  $F$  at each snapshot  $r$  from the snapshot set  $R$  and gives the total flow setup time in the system for one hour named as  $A_f$ .

$$A_f = \sum_{f \in F} \sum_{r \in R} d_{r,f[src],f[dst]} + 2 \cdot d_{r,s,c} \cdot \sum_{f \in F} \sum_{r \in R} \sum_{c \in C} \sum_{s \in p_{f,r}} [k_{s,f} \cdot x_{s,c} + x_{s,c} - z_{c,s,k}] \quad (2)$$

TABLE I  
OPTIMIZATION INPUT PARAMETERS AND DECISION VARIABLES

Input Parameters	
$S$	Set of satellites
$V$	Set of nodes (satellites, gateways)
$E$	Set of links where $E \subseteq V \times V$
$C$	Set of controllers where $C \subseteq S$
$F$	Set of the flows in the network, where $f[src]$ and $f[dst]$ are the source and destination for $f \in F$
$R$	Set of the snapshots of the network
$k_{s,f}$	constant variable indicating whether satellite $s \in S$ is the initiator of the flow $f \in F$
$p_{f,r}$	Set of nodes $V$ which are part of the flow path from source to destination of $f \in F$ , $r \in R$
$K$	number of controllers to be placed
$d_{r,s,k}$	forwarding latency from satellite $s$ to satellite $k$ at snapshot $r$ , where $s, k \in S$
List of variables	
$y_c$	binary variable notating if a controller is place on $c \in C$
$x_{s,c}$	binary variable notating if a node $s \in S$ is assigned to $c \in C$
$z_{c,s,k}$	binary variable notating if two consecutive nodes $s, k \in S$ are assigned to the same controller $c \in C$

### D. Optimization problem formulation

The dynamic controller placement is formulated as an ILP and is solved in a Gurobi framework implemented in Python. The input parameters for this optimization problem along with the optimization variables are denoted in table I.

The optimization goal is to minimize the average flow setup time for a given flow profile  $F$ .

$$\min \frac{1}{|F|} (A_f) \quad (3)$$

The constraints related to the controller placement problem are as follows:

Constraint (4), which assures that the total number of controllers to be placed in the network is  $K$ .

$$\sum_{c \in C} y_c = K, \quad (4)$$

Constraint (5), guarantees that a satellite  $s$  is controlled by a controller  $c$  only if the controller is active.

$$x_{s,c} \leq y_c, \forall s \in S, \forall c \in C \quad (5)$$

Constraint (6), ensures that each satellite  $s$  is controlled by exactly one controller.

$$\sum_{c \in C} x_{s,c} = 1, \forall s \in S \quad (6)$$

If two satellites belong to different controller domains they need to be assigned to different controllers. A helper binary variable  $z_{c,s,k}$  is identified to quantify such an occurrence. Constraint (7) guarantees such an assignment if two satellites belong to different clusters.

$$z_{c,s,k} = x_{s,c} \cdot x_{k,c}, \forall s \in S, \forall k \in S, \forall c \in C \quad (7)$$

In order to be able to solve the problem in a linear optimizer, constraint (7) is replaced by three equations.

$$z_{c,s,k} \leq x_{s,c}, \forall s \in S, \forall k \in S, \forall c \in C \quad (8a)$$

$$z_{c,s,k} \leq x_{k,c}, \forall s \in S, \forall k \in S, \forall c \in C \quad (8b)$$

$$z_{c,s,k} = x_{s,c} + x_{k,c} - 1, \forall s \in S, \forall k \in S, \forall c \in C \quad (8c)$$

#### IV. SYSTEM PERFORMANCE EVALUATION

##### A. Dynamic Controller Placement

We study initially the dynamic controller placement in a LEO constellation. For this evaluation we consider a daily simulation starting from 7/25/2017 10 AM GMT up to 7/26/2017 9 AM GMT. Every one hour, the optimization problem is applied in order to decide the controller selection and satellite-to-controller assignment. For each of the hours, the number of controllers  $K$  is varied in order to observe the impact on the average flow setup time. The results of this operation are illustrated in Fig. 5. Considering the synchronization cost between the controllers in a distributed control plane, a tolerable synchronization overhead must be identified in order to decide the adequate number of controllers to be deployed in the network [20]. Therefore, we examine the average flow setup time up to 10 controllers. As the number of controllers  $K$  is increasing the average flow setup time decreases for all the simulation hours. It can be observed that during 12 AM GMT at midnight in Europe the average flow setup time achieved for all the number of controllers  $K$  is minimal, whereas the maximum is achieved 6 PM GMT where the user activity is higher in Europe in accordance to the traffic model.

##### B. Dynamic vs. Static Controller Placement

In order to evaluate the benefits of the dynamic controller placement, we focus on the comparison with the static approach. Differently from the dynamic approach, in a static approach the controller placement and satellite-to-controller selection remains unchanged. During our work we select as a static controller set the satellites most frequently selected as controllers, taking into account all the simulation hours. In order to determine the static controller selection, a probability color map is created. This color map, denotes the probability of each satellite being selected as a controller within a day.

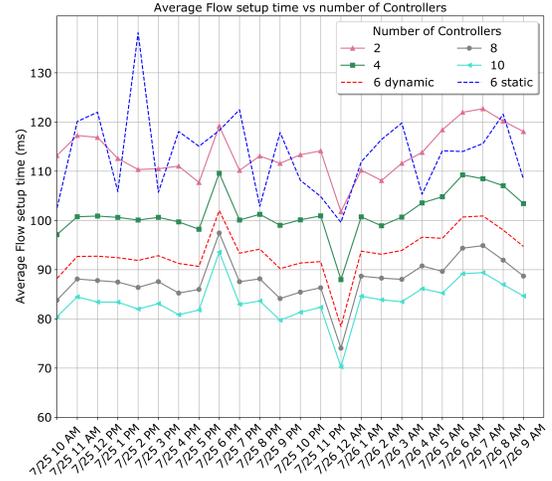


Fig. 5. Optimal average flow setup time  $ms$  for various simulation GMT hours as a function of the number of controllers  $K$ . Furthermore the average flow setup time is illustrated for the static approach with blue dotted line for  $K = 6$ .

This operation is realized for various controllers deployed in the network. For the comparison considered in this paper, the number of controllers  $K$  is set to 6. Fig. 5 illustrates the results regarding the average flow setup time. It is observed that the dynamic approach outperforms the static approach for all the simulation hours. The performance difference varies between 8.5% and 40%. However on average there is 20% difference in the performance towards the dynamic approach. The results demonstrate that the static controller placement can cause severe issues for traffic fluctuations, therefore a dynamic adaptation becomes vital. Nonetheless, extra migration costs occur for the dynamic approach. Therefore, a trade-off must be found when developing the system. However, the migration cost is out of the scope of this paper.

##### C. Architecture Comparison

Besides the proposed architecture in this paper, two alternative solutions exist in the literature. The study in [14] suggests the SDN controller placement on the SG, whereas [12] proposes the controller placement on the GEO satellites. However, none of the proposed works target the controller placement problem. In order to be able to evaluate and compare the two systems, we leverage the traffic model and ILP proposed in our paper.

For each of the aforementioned architectures, we consider each simulation hour as a single output of the optimization problem. The 24 outputs are then grouped together and boxplots are created in order to clearly observe the average flow setup variations. The number of controllers  $K$  is varied and the results are illustrated in Fig. 6. As shown in the figure the following observations can be made: 1) For the proposed SDN-enabled LEO constellation, it is clear that the average flow setup time monotonically decreases with the increase of  $K$ , however the gain is not linear. The law of diminishing returns is identified since at some point adding more controllers results in minor benefits. 2) For the architecture where the SG are selected as controllers, regardless of the

number of controllers  $K$  the average flow setup time is not decreasing and it reaches fast a saturation point. Due to the fact that geographically they are close to each other, deploying more controllers does not bring any benefit. 3) SG-controllers achieve better results than LEO-controllers up to 6 controllers deployed in the network. However, while  $K$  increases the difference decreases. From 7 controllers and on, LEO-controllers shows better performance. The highest difference is achieved for 10 controllers where LEO-controllers outperforms the SG-controllers by achieving 5% better results for the median and 13% better results for the 75 percentiles. Further increasing the number of controllers can certainly improve the performance, however at higher operational expenditures, hence making the configuration not significant from a deployment viewpoint.

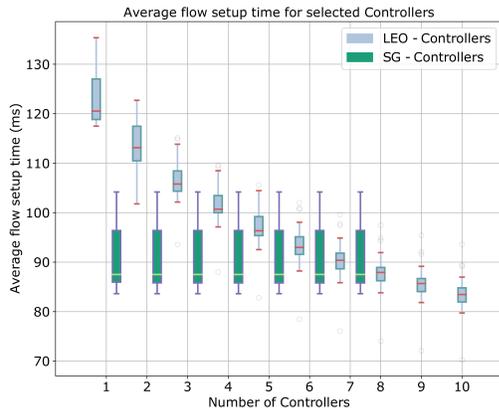


Fig. 6. Optimal average flow setup time  $ms$  for different architecture as a function of the number of controller  $K$ .

Due to the high altitude of a GEO satellite, the forwarding latency towards the satellites of the LEO constellation which is more than 100  $ms$  effects drastically the average flow setup time. These values vary between 320  $ms$  up to 335  $ms$ , more than twice of the values recorded for the LEO-based SDN architecture. Hence, full analysis of the results for the GEO-based architecture is omitted.

## V. CONCLUSIONS

In this paper we investigated the use-case of an SDN-enabled LEO constellation. We provided a traffic model considering user demands depending both on time and user's geographical position. Moreover, we proposed an ILP to solve the dynamic controller placement problem in the LEO constellation while minimizing the average flow setup time. Initial results were provided regarding the dynamic controller placement in a dynamic satellite network and we showed that the performance provided by the dynamic approach outperforms the static approach. Finally, we leveraged our model to evaluate and compare our proposed SDN approach with other existing approaches and we showed that the proposed SDN-enabled LEO constellation architecture is more promising than the SG-controller approach when more than 6 controllers are deployed in the network.

As a future work we intent to extend our approach by considering the migration cost and communication overhead

among the controllers and provide a joint optimization problem for the dynamic controller placement.

## REFERENCES

- [1] ETSI, "Satellite earth stations and systems (ses); combined satellite and terrestrial networks scenarios." Tech. Rep. 103 124, 2013.
- [2] R. Ferrús, H. Koumaras, O. Sallent, G. Agapiou, T. R. M., A. Kourtis, C. Boustie, P. Gelard, and T. Ahmed, "SDN/NFV-enabled satellite communications networks: opportunities, scenarios and challenges," *Phys. Commun.*, vol. 18(2), pp. 95–112, 2016.
- [3] L. Bertaux, S. Medjah, P. Berthou, S. Abdellatif, A. Hakiri, P. Gelard, S. Planchou, and M. Bruyere, "Software defined networking and virtualization for broadband satellite networks," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 54–60, Mar. 2015.
- [4] A. Guidotti, A. Vanelli-Coralli, M. Caus, J. Bas, G. Colavolpe, T. Foggi, S. Cioni, A. Modenini, and D. Tarchi, "Satellite-enabled LTE systems in LEO constellations," in *Communications Workshops (ICC Workshops), 2017 IEEE International Conference on*. IEEE, 2017, pp. 876–881.
- [5] Y. Ganjali and A. Tootoonchian, "HyperFlow: A Distributed Control Plane for OpenFlow." in *INM/WREN*, 2010.
- [6] S. Hassas Yeganeh and Y. Ganjali, "Kandoo: a framework for efficient and scalable offloading of control applications," in *Proceedings of the first workshop on Hot topics in software defined networks*. ACM, 2012, pp. 19–24.
- [7] M. He, A. Basta, A. Blenk, and W. Kellerer, "Modeling flow setup time for controller placement in SDN: Evaluation for dynamic flows," in *Proc. IEEE Int. Conf. Communications (ICC)*, May 2017, pp. 1–7.
- [8] M. F. Bari, A. R. Roy, S. R. Chowdhury, Q. Zhang, M. F. Zhani, R. Ahmed, and R. Boutaba, "Dynamic Controller Provisioning in Software Defined Networks," in *Proc. 9th Int. Conf. Network and Service Management (CNSM 2013)*, Oct. 2013, pp. 18–25.
- [9] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," in *Proceedings of the first workshop on Hot topics in software defined networks*. ACM, 2012, pp. 7–12.
- [10] S. Lange, S. Gebert, T. Zinner, P. Tran-Gia, D. Hock, M. Jarschel, and M. Hoffmann, "Heuristic approaches to the controller placement problem in large scale SDN networks," *IEEE Transactions on Network and Service Management*, vol. 12, no. 1, pp. 4–17, 2015.
- [11] P. Vizarrera, C. Mas Machuca, and W. Kellerer, "Controller placement strategies for a resilient SDN control plane," in *Resilient Networks Design and Modeling (RNDM), 2016 8th International Workshop on*. IEEE, 2016, pp. 253–259.
- [12] J. Bao, B. Zhao, W. Yu, Z. Feng, C. Wu, and Z. Gong, "OpenSAN: a software-defined satellite network architecture," in *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4. ACM, 2014, pp. 347–348.
- [13] T. Li, H. Zhou, H. Luo, Q. Xu, and Y. Ye, "Using SDN and NFV to Implement Satellite Communication Networks," in *Proc. Int. Conf. Networking and Network Applications (NaNA)*, Jul. 2016, pp. 131–134.
- [14] P. Du, S. Nazari, J. Mena, R. Fan, M. Gerla, and R. Gupta, "Multipath TCP in SDN-enabled LEO satellite networks," in *Proc. MILCOM 2016 - 2016 IEEE Military Communications Conf*, Nov. 2016, pp. 354–359.
- [15] E. Lutz, M. Werner, and A. Jahn, *Satellite Systems for Personal and Broadband Communications*. Springer, 2000.
- [16] C. Chen and E. Ekici, "A routing protocol for hierarchical LEO/MEO satellite IP networks," *Wireless Networks*, vol. 11, no. 4, 2005.
- [17] AGI, "Systems Tool Kit (STK)," <https://www.agi.com/products/engineering-tools>.
- [18] X. Alberti, J. M. Cebrian, A. D. Bianco, Z. Katona, J. Lei, M. A. Vazquez-Castro, A. Zanus, L. Gilbert, and N. Alagha, "System capacity optimization in time and frequency for multibeam multi-media satellite systems," in *2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop*, Sept 2010, pp. 226–233.
- [19] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.
- [20] E. Sakic and W. Kellerer, "Response Time and Availability Study of RAFT Consensus in Distributed SDN Control Plane," *IEEE Transactions on Network and Service Management*, vol. 15, no. 1, pp. 304–318, 2018.