

# Evaluating Software-Defined Networking-Driven Edge Clouds for 5G Critical Communications

Fabian Kurtz, Igor Laukhin, Caner Bektas, Christian Wietfeld

Communication Networks Institute, TU Dortmund University, Otto-Hahn-Strasse 6, 44227 Dortmund

Email: {fabian.kurtz, igor.laukhin, caner.bektas, christian.wietfeld}@tu-dortmund.de

**Abstract**—Modern societies are increasingly dependent on Critical Infrastructures (CIs) such as Smart Grids (SGs) and Intelligent Transportation Systems (ITS). Due to continuously rising demands in terms of efficiency and capabilities, complex control and monitoring strategies are employed. These in turn rely on robust, high performance communication networks. As CIs are distributed systems with diverging demands, the deployment of individual networks is often too costly and time consuming. Furthermore, the vast physical scope traditionally results in unacceptably high end-to-end delays. Therefore, a convergence of public and dedicated networks, as well as traditional Information and Communication Technology (ICT) with Information Technology (IT), are considered key aspects of 5G for addressing these challenges. Hence, this paper provides an empirical evaluation of CI communication services on basis of a Software-Defined Networking (SDN) and Network Function Virtualization (NFV) driven Edge Clouds (ECs) within a sliced 5G network. By shifting computing resources from the backbone or back office towards the network’s access level, ECs allow for drastically reduced delays. Also, the traffic load on several layers of the communication infrastructure is reduced, as data can be kept locally, i.e. close to the source. This is demonstrated by shifting an ITS application from a central cloud to the EC. Services are transferred step-wise and transparently, minimizing interruptions while dynamically adapting to the backhaul’s available data rate. The developed system is evaluated under realistic traffic conditions within a physical testing environment.

## I. INTRODUCTION

The ongoing shift to renewable energies, e.g. photovoltaics, water or wind turbines, marks the transition to SGs. Similarly, the emergence of connected and autonomous cars contributes to the creation of ITS. These CIs place exacting demands in terms of e.g. data rate, delays and robustness against failures or attacks on the underlying communication system. To achieve this performance while fulfilling the required Quality of Service (QoS) level on a single underlying infrastructure, 5G proposes network slicing. This technology, for which no standard currently exists, allows the instantiation of virtual networks on top of a shared physical infrastructure. Thereby, such Ultra-Reliable Low Latency Communication (uRLLC) communications can utilize public networks, reducing costs and deployment times. However, due to the large, spatial scope of e.g., Smart Grids, data traffic has to traverse great distances and multiple infrastructure layers. This is shown in the lower section of Figure 1, where devices of the different use cases transmit their data for processing to the back office or main cloud. En route, access, aggregation and core networks have

to be passed. The same applies to the calculated responses, therefore delays and thus reaction times are increased greatly. Furthermore, the vast number of devices (e.g. cars, smart meters) results in an amount of traffic which may exceed network capacity. Hence, this work proposes and analyses SDN and NFV driven ECs. By processing data at a network’s boundary, i.e., edge, end-to-end delays and core network utilization are drastically reduced, as shown in the upper part of Figure 1. Depending on the applications, robustness is also improved, as failures in the backhaul do not affect services deployed on an EC. The proposed solution approach integrates network slicing with ECs to meet the specific requirements of CI communications. SDN serves as orchestrator, detecting potential overloads and mitigating them by transparently instantiating the appropriate Virtual Network Functions (VNFs) on an EC. We present a comprehensive evaluation on basis of a realistic ITS scenario. Through implementation in a physical testing setup, meaningful performance data is provided. The paper is structured as follows: First, an overview of related work is given in Section II. Afterwards, Section III discusses selected fundamentals of the employed technologies and methodologies. There, details of the developed EC solution are also introduced. Next, Section IV provides a description of the employed evaluation scenario and physical testing environment. Performance figures and results obtained on this basis are laid out in Section V. Lastly, a conclusion and an outlook on future work complete this paper.

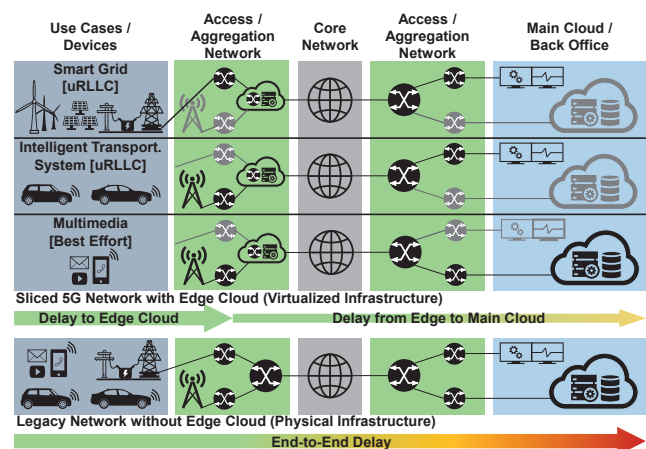


Figure 1: Edge Clouds in Sliced 5G Communication Networks

## II. RELATED WORK

Related work can be categorized into two domains: EC centric studies and those which focus on 5G network slicing. Particularly relevant for the purposes of this paper, are works which address network slicing in wired communication infrastructures. Here, the surveys [1], [2], [3], [4] and [5] agree on the importance of SDN as well as NFV for implementing such solutions. A comprehensive review of potential architectural concepts is also provided. Generally, extensions of the OpenFlow (OF) proxy FlowVisor [6] are regarded as possible approaches to network slicing and studied via experimental evaluations. In this context, management [7], security [8] as well as basic extensions to enable QoS [9] are implemented and examined. In contrast, the system utilized in this work achieves traffic shaping, i.e. inter-slice QoS, and slice isolation based on Hierarchical Token Bucket (HTB) queues. Other prototypes of network slicing systems employing the OpenDaylight controller [10] and are described in [11] and [12]. Finally, works [13], [14], [15] and [16], [17] as well as [18] conduct algorithmic, respectively theoretical studies. Yet, many of these works concentrate on optimizing the chaining of network functions within a slice, rather than enabling hard service guarantees. However, the latter is exceptionally relevant for CI communications such as those within SGs and ITS. Out of such reasoning this paper's slicing solution is evaluated within a physical testing setup and on basis of a realistic use case scenario derived from real-world applications.

For ECs, also known as Edge Computing, the current state of the art is e.g. surveyed by [19]. A widely discussed application is the use of EC for content distribution and caching. E.g. [20] present an edge service orchestration platform for webpages, while others focus on optimization of established caching strategies. This contrasts with the presented work, as CIs are of great priority, highly delay sensitive and - due to their interactive nature - rarely profit from caching. The work detailed in [21] utilizes SDN to provide ECs for Industrial Internet of Things (IoT) environments. In the case of wireless IoT sensors, energy consumption is a crucial factor. Therefore, the authors strive to strike a balance between energy efficiency and latency while maintaining the desired QoS level. While insightful, the work is limited to an analytical evaluation. Similarly, [22] survey different cloud technologies for enabling future IoT applications. In consequence, a good overview is given, while no quantitative results are achieved. Migrating running services from a main cloud or back office server to an EC is studied by [23]. Different, pre-existing state migration schemes are contrasted with a novel strategy on the example of a latency sensitive game. Meanwhile, as e.g. SGs lack mobility applications can be pre-deployed at a network's edge nodes, effectively avoiding such delays entirely. In summary, the combination of a challenging real-world application such as CI communication, network slicing and SDN-driven ECs, is a relevant field of study. Yet, empirical performance observations are mostly limited to a subset of technologies. Hence, we focus on a comprehensive implementation and evaluation.

## III. SOFTWARE-DEFINED NETWORKING AND NETWORK FUNCTION VIRTUALIZATION BASED 5G EDGE CLOUDS

This section provides a brief discussion of the requirements imposed by CIs. Afterwards, the developed Edge Cloud architecture and its underlying technologies are introduced.

### A. Critical Infrastructure Communication Requirements

Due to the critical importance of their services CIs define strict requirements in terms of the underlying ICT. Particularly exacting demands are defined in the International Electrotechnical Commission (IEC)'s standard 61850 [24]. It is crucial for substation monitoring, control and protection services as the share of fluctuating, renewable power generation in SGs rises. The message types Sampled Values (SV) and Generic Object Oriented Substation Events (GOOSE) are employed for sending measurements and events, respectively. Both are directly encapsulated into IEEE 802.3 Ethernet frames, with packet sizes between 64 and 1518 Bytes. Depending on the service, maximum end-to-end latencies range from 3ms to 20ms, regardless of failures, with 4,000 to 12,800 packets per second. Hence, SG traffic can be categorized as uRLLC. ITS provide the main evaluation scenario of this work, constituting a safety-critical infrastructure. As vehicles become increasingly connected and road traffic shifts towards higher degrees of automation, more Floating Car Data (FCD) is generated. It contains status data including vehicle speed, location or direction of travel. This information can be shared among vehicles or be transmitted to Roadside Units (RSUs). From there it is transferred to back offices, allowing for analysis and road traffic optimization to e.g. reduce accidents. Packet sizes, formats, etc., are not yet standardized. Hence, we resort to values from literature [25]. Due to their requirements, ITS and SGs stand to benefit greatly from slicing and ECs.

### B. An NFV and SDN driven Approach to Edge Clouds

ITS as well as SGs are dependent on pervasive, robust communication infrastructures. As dedicated networks are associated with high costs and lengthy deployment times, slicing is considered a suitable alternative. Through virtualization of

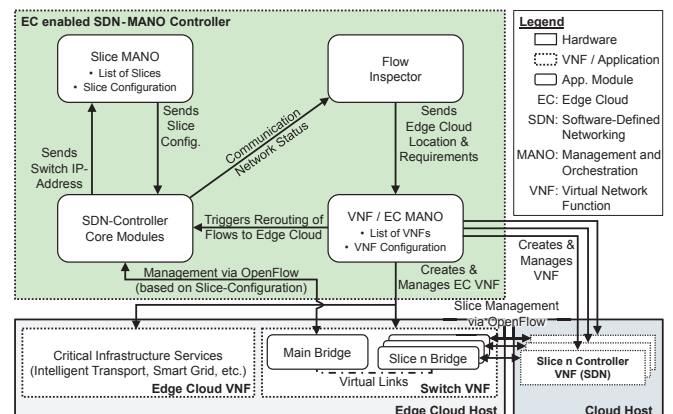


Figure 2: The Developed Software-Defined Networking Driven 5G Edge Cloud Architecture

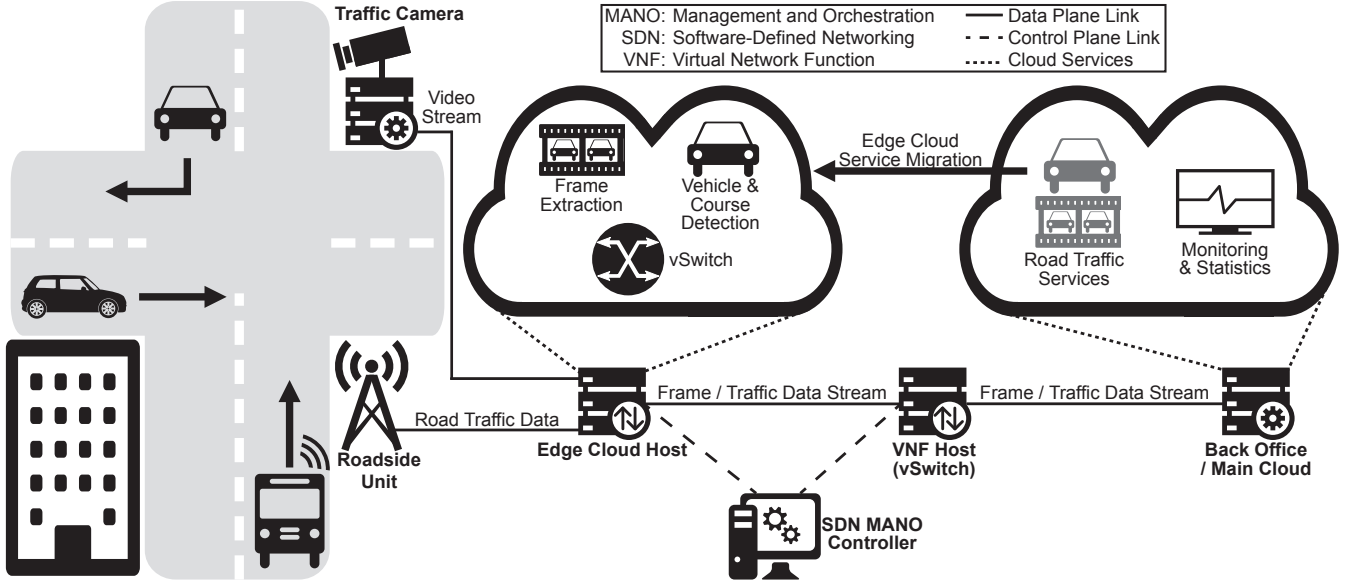


Figure 3: Testing Setup for Software-Defined Networking Driven Edge Clouds in Intelligent Transportation Systems

network resources, it enables traffic isolation, hard service guarantees and flexible generation of necessary topologies. In the scope of this paper, we focus on enabling the two former aspects in backhaul networks, by combining SDN and NFV. SDN decouples the control plane (e.g. determination of routes) from physical packet forwarding, i.e. the data plane. Thereby, the decision process is centralized in so-called SDN controllers, which configure forwarding elements (i.e. switches) via the de-facto standard OF protocol [26]. NFV follows a similar paradigm, by shifting traditionally fully integrated functionalities from hard- to software components. However, it is broader in scope as network services such as firewalls, load balancers, intrusion detection systems and more are virtualized. Thus, such VNFs can be run on Commercial Off-The-Shelf (COTS) server hardware, allowing for their flexible deployment and upgrade independent of physically changing the communication infrastructure. Both SDN and NFV also form the basis for the proposed EC for CI communication architecture, which is depicted in Figure 2. Our Software-defined Universal Controller for Communications in Essential SystemS (SUCCESS) controller, originally forked from Floodlight [27] and specifically tailored to CI requirements, is enhanced with slicing [28] and EC capabilities. It provides Management and Orchestration (MANO) functionalities to dynamically deploy VNFs across the communication infrastructure's host platforms. Slices are created, by establishing dedicated, per slice queues, thus enabling stable traffic prioritization. These virtual infrastructures are each being handled by their own controller, allowing for independent configuration by their respective owners. Further details of the employed slicing approach can be found in [28]. For providing EC services, new and existing traffic flows are continuously monitored by a flow inspector module. If a VNF capable of processing the flow exists, the host closest to the traffic source is located. Based

on this data, the VNF MANO module starts and configures the appropriate, virtualized service on the EC host. Also, a re-routing of the affected flow to this new destination is triggered. Thereby, shifting data processing from the back office or main cloud to an EC is fully transparent to the client application. Through this, load on the backhaul is reduced, freeing up network resources. Hence, failures or overloads in the core have no impact on CI services running on ECs. Further, use cases can scale beyond the backhaul's capability to carry traffic, or data rate allocated for slices can be reduced, lowering costs for leasing network capacity. Another benefit are decreased delays, enabled by the reduced physical distance between data sources / sinks and processing. The ability to flexibly instantiate services is another crucial factor, affording network users the option to adapt, scale and react rapidly to changes in a use case's demand or application profile. Thus, the requirements of CI operators without dedicated communication infrastructure, as given in the previous section, are addressed on shared, public networks. Moreover, traffic flows are sent via slices, according to their priority. This is possible, as traditional switches have been replaced by COTS hosts, on which virtualized data planes are deployed as VNFs. SDN controllers handling individual slices are also realized as VNF. Thereby, a fully virtualized communication infrastructure is created. In consequence, this concept follows the paradigm shift towards softwareized networks, which is a foundation of future 5G networks [29]. Overall, it can be said that ECs are a key component to achieving the 1ms end-to-end delay requirement as defined by 5G and contribute to the evolution of communication networks beyond the classical role of purely transmitting data. An example for utilizing the proposed EC for CI communication solution in the context of ITS is given by Figure 3. It serves as basis for our empirical evaluation and is described in detail in the following Section.



#### IV. TESTING ENVIRONMENT AND EVALUATION SCENARIOS

The scenario and testing setup employed for evaluating the proposed EC solution are introduced in the following sections.

##### A. Physical Testing Environment

Components of the physical testing setup are as follows. Five identical COTS servers, each equipped with an Intel Xeon D-1518 Central Processing Unit (CPU) (four 2.2GHz cores), 16GB of RAM and six 1GBase-T Ethernet ports via two Network Interface Cards (NICs) (two: Intel I210, four: Intel I350). Ubuntu Server 16.04.3 LTS (v4.13.0-32-generic x86-64 Kernel) is used as Operating System (OS). As depicted by Figure 3, one device acts as video traffic source, with another server configured as back office / main cloud host. Two computers form the sliced communication network, with one hosting the EC and the other tasked with switching. The SDN MANO (Ryu v4.19 [30]) and slice controllers (SUCCESS) are deployed on the remaining device. A separate out-of-band control network (dashed lines) avoids interference of data and management planes. Virtual switches utilize Open vSwitch (OVS) (v2.5.2) [31], while the analyzed ITS use case employs OpenCV (v2.4.8), Flask (v1.0.2) and OpenALPR (v2.3.0). All software components are deployed as container-based (Docker 18.03.1-CE) VNFs. For studying slicing, User Datagram Protocol (UDP) traffic is generated with iperf2 (v2.0.10 [32]) to recreate real-world communication patterns.

##### B. Smart Intersection Based Evaluation Scenario

Figure 3 shows an ITS' urban smart intersection, as implemented and evaluated within the testing environment. While connected or autonomous cars commonly have the ability to communicate either directly or via RSUs with other vehicles, this option is currently not available to all road users. For example, the depicted Bus may not be aware of the conventional car on the left, due to an obstructed line-of-sight. By utilizing cameras with a full view of the crossing, data for avoiding accidents is gathered and used for informing vehicles or switching traffic lights. We replicate the camera's output by streaming a previously captured 3840\*2160 Pixel video at 60 Hz with a mean bit rate of 32.76 Mbps. It is sent to the back office for processing, thereby determining position, number and the road traffic's vector of travel. Results can be sent to the RSU (not included in the test setup) to inform vehicles of the intersection's status and obstacles. Once the EC is activated, the SDN controller detects and shifts the flow to the host closest to the data traffic source. There it can be partially processed by extracting and sending video key frames to the back office, thus reducing backhaul load. In case a further reduction in data rate is desired, the entire processing (e.g. vehicle detection, etc.) is performed within the EC. Hence, failures of the back office or backhaul have no impact on the ITS, as these network parts are only relevant for statistics collection instead of critical road safety. The evaluation focuses on end-to-end and VNF instantiation delays as well as core network load in a sliced EC-driven network.

#### V. EVALUATION RESULTS

In this Section, a comprehensive evaluation of the developed EC is given within a sliced CI communication network.

##### A. Slicing for Critical Infrastructure Communications

Table I provides an overview of slices deployed across the implemented testing setup depicted by Figure 3. Three slices are used to isolate ITS, SGs and other best effort data traffic from one another. Both CIs are categorized as uRLLC 5G services, with hard minimum data rate guarantees of 50 respectively 30 Mbps. FCD and IEC 61850 are used for recreation of realistic CI application traffic. Figure 4 shows results obtained on basis of these parameters for a single physical link of the backhaul. The plot is divided into the left part, without slicing, and the right section, with slicing. In both cases, best effort traffic is sent with the maximum link data rate. Once ITS packets enter the network at 10s, both applications compete for resources. Accordingly, a fluctuation of both flows is observed, while the more critical FCD does not achieve the desired 50 Mbps. Shortly afterwards, at around 20s SG IEC 61850 packets are also transmitted via the link under test. This service also does not achieve its required minimum data rate, despite being of higher priority. Best effort as well as the critical ITS service lose some link capacity to the newly added flow. Thus, such a setup would not be suitable to meet the demands of CI communications. Activating slicing mitigates these issues, as shown on the figure's right. At ~45s an ITS flow starts, stably achieving its guaranteed bit rate. Lower priority best effort packets only receive link access once all higher priority slices are addressed. The same behavior is observed at ~55s, with stable SG power line protection traffic. Requirements are fulfilled and only the lowest priority service, i.e. best effort, is affected. It is to be noted, that an appropriate assignment of network resources to slices is assumed (c.f. [28]). Higher priority slices like CIs may displace those of lower priority, if it is the only option to fulfil guarantees. This is to be considered when dimensioning slices and defining Service Level Agreements (SLAs). All subsequent results have been obtained within an ITS slice, providing appropriate prioritization and isolation of this critical infrastructure's data.

##### B. Edge Cloud Enabled Communication Networks

First, we study the shift from main to edge clouds, for processing of the intersection's video feed. Figure 5 gives the delays incurred for the individual steps of this action. The SDN controller's monitoring takes a mean of 2.7 s (median:

Table I: Slices and Traffic Flows of the CI Comm. Scenario

Slice	Use Case	5G Service Class	Hard Min. Data Rate [Mbps]	Max. Delay [ms]
Intelligent Transportation Systems	Floating Car Data	uRLLC	50	1
Smart Grids	Protection (IEC 61850)	uRLLC	30	1
Best Effort	Multimedia	None	None	100

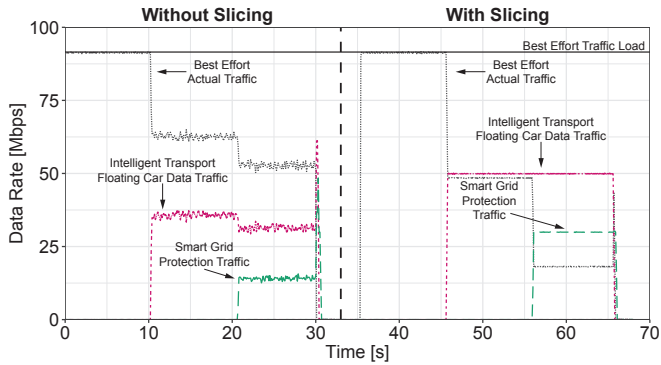


Figure 4: Data Rate Allocation Among Critical Infrastructure Communication Service without and with Network Slicing

2412 ms) to detect network congestion (e.g. caused by adding another video stream to an inadequately configured slice). This value mainly results from the 1 s network statistics polling frequency, constituting the lowest value accepted by our switches. Depending on when measurements occur relative to the start of the congestion, two CDF peaks result (c.f. first violin). Also, two measurement samples are used to avoid deploying ECs for flows with only short bursts in data rate. Subsequently, or if the move to an EC is triggered manually, the SDN MANO controller starts the pre-deployed VNF on the host, which it identifies as closest to the traffic source. After 764 ms (median: 762 ms) the start-up phase is completed. Next, existing routes are recalculated. This is implemented to optimize network utilization for those flows, which previously shared links with ITS traffic. Also, the virtualized switch on the EC host is configured to interface with the newly created VNF. These actions take 1208 ms (median: 1211 ms) to complete. After another 41 ms (median: 33 ms) calculated routing updates are installed on the network's switches to forward the video stream to the EC for processing. Thus, the transition is completed. Outliers are a result of the employed SDN controller, which is based on the Java programming language. Its so-called garbage collection causes intermittent pauses in which computer memory allocation is managed, contributing to the observed delay spikes. Due to the design of

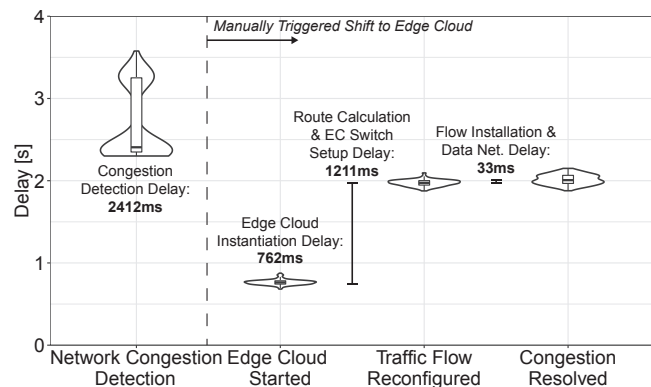


Figure 5: Delay Breakdown for e.g. Mitigating Network Congestion by Transitioning to Edge Cloud based ITS Services

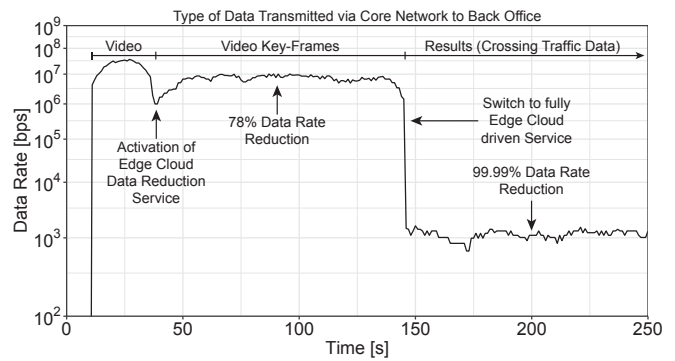


Figure 6: Data Rate Reductions in Backhaul Networks via Incremental Main to Edge Cloud Transition for ITS Services

the studied ITS service, it is possible to only partially offload processes to the EC. This is illustrated by Figure 6, which highlights the resulting, significant reductions in backhaul data rate requirements. Initially, the entire video stream of the smart urban intersection is sent to the main cloud, i.e. back office. By activating the EC at 38 s, the VNF extracts key frames, compresses, and sends them for processing across the network. This reduces the average utilized data rate from 32.76 Mbps to 7.28 Mbps. Another option is to extract the relevant data (e.g. direction of the vehicles' travel) directly within the EC. Hence, at 145 s only status reports are sent to the back office, while the ITS service itself is fully EC-based. Effectively, a further reduction in data rate to just 0.01% is achieved, with just 1.16 kbps of non-critical backhaul traffic. While the first option, i.e. transmitting compressed key frames, reduces network load, no end-to-end delay gains can be achieved. Therefore, the most sensible action to fulfill the ITS use case's requirements, is to rely on a fully EC-based service. In this case, vehicles and traffic lights receive intersection status updates more quickly. As the video stream is processed in close physical proximity to road users, factors which contribute to end-to-end delay (e.g. signal propagation, network hops) are reduced. Figure 7 illustrates this, by contrasting results measured for running the ITS service with and without EC. Despite the limited scope of our physical testing setup, delay is reduced by 110  $\mu$ s.

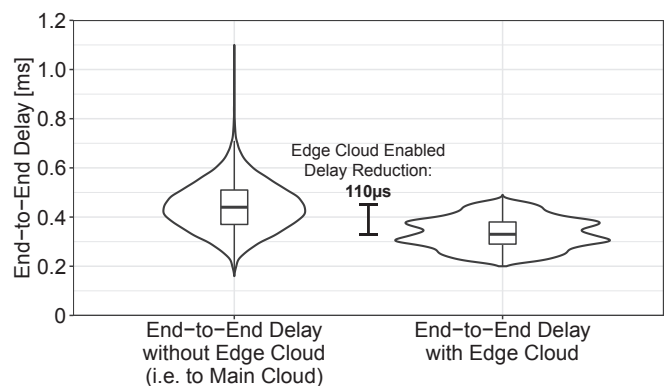


Figure 7: End-to-End Delays for Main and Edge Cloud based Critical Infrastructure Communication Services (two Hops)

Critically, without an EC some outliers exceed the 1 ms mark. This is not acceptable for 5G communication networks, as this leaves no headroom for adding an air interface (in this case an RSU). In contrast, our proposed EC solution utilizes just short of 0.5 ms in the worst case, leaving roughly the same time for radio transmission. It is to be noted, that all measurements are repeated a minimum of 50 times to achieve sufficient statistical confidence. Moreover, testing equipment is synchronized via the Precision Time Protocol (PTP) [33], with a mean clock deviation of 16  $\mu$ s and 152  $\mu$ s at maximum.

## VI. CONCLUSION AND OUTLOOK

This work introduces an approach to EC-based CI communication in shared 5G infrastructures. By harnessing SDN and NFV, we develop an EC MANO scheme, that is integrated with network slicing. A real-world application scenario is derived on basis of the exacting performance requirements of SGs and ITS. This is then empirically evaluated in a laboratory testing setup. Results show a significant reduction in backhaul traffic as well as end-to-end delay by shifting to EC-based services. Meanwhile, network slices are shown to enable hard service guarantees, achieving the desired QoS in terms of data rate. Transition delay between main and edge cloud is minimized, allowing the shift to occur without affecting applications. Due to an optimized SDN-centric control process, EC deployment is shown to be fully transparent to end users. Future work will aim to provide a fully 5G compliant solution, by integrating an EC-based mmWave radio. Moreover, implementation in scope of a physical SG demonstrator is targeted. Thereby results would be verified within another CI, based on physical components of the electrical grid.

## ACKNOWLEDGMENT

This work has been carried out in the course of the Franco-German Project *BERCOM* (FKZ: 13N13741), co-funded by the Agence Nationale de la Recherche (ANR) and the German Federal Ministry of Education and Research (BMBF).

## REFERENCES

- [1] P. Rost *et al.*, 'Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks', *IEEE Communications Magazine*, vol. 55, no. 5, pp. 72–79, May 2017.
- [2] J. Ordonez-Lucena *et al.*, 'Network Slicing for 5G with SDN/NFV: Concepts, Architectures, and Challenges', *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80–87, May 2017.
- [3] H. Zhang *et al.*, 'Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges', *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, 2017.
- [4] T. Soenen, R. Banerjee, W. Tavernier, D. Colle and M. Pickavet, 'Demystifying network slicing: From theory to practice', in *IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, May 2017, pp. 1115–1120.
- [5] X. Foukas, G. Patounas, A. Elmokashfi and M. K. Marina, 'Network Slicing in 5G: Survey and Challenges', *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, May 2017.
- [6] R. Sherwood *et al.*, 'FlowVisor: A Network Virtualization Layer', Tech. Rep., Oct. 2009.
- [7] J. L. Chen, Y. W. Ma, H. Y. Kuo and W. C. Hung, 'Enterprise visor: A Software-Defined enterprise network resource management engine', in *IEEE/SICE International Symposium on System Integration*, Dec. 2014, pp. 381–384.
- [8] C.-H. Chen, C. Chen, S.-H. Lu and C.-C. Tseng, 'Role-based campus network slicing', in *International Conference on Network Protocols (ICNP)*, Nov. 2016, pp. 1–6.
- [9] C. W. Tseng *et al.*, 'A network traffic shunt system in SDN network', in *International Conference on Computer, Information and Telecommunication Systems (CITS)*, Jul. 2017, pp. 195–199.
- [10] *OpenDaylight*, 2018. [Online]. Available: [www.opendaylight.org](http://www.opendaylight.org).
- [11] X. Li *et al.*, '5G-Crosshaul Network Slicing: Enabling Multi-Tenancy in Mobile Transport Networks', *IEEE Communications Magazine*, vol. 55, no. 8, pp. 128–137, 2017.
- [12] P. K. Chartsias *et al.*, 'SDN/NFV-Based End to End Network Slicing for 5G Multi-Tenant Networks', in *European Conference on Networks and Communications (EuCNC)*, Jun. 2017, pp. 1–5.
- [13] S. Vassilaras *et al.*, 'The Algorithmic Aspects of Network Slicing', *IEEE Communications Magazine*, vol. 55, no. 8, pp. 112–119, 2017.
- [14] R. Trivisonno, R. Guerzoni, I. Vaishnavi and A. Frimpong, 'Network Resource Management and QoS in SDN-Enabled 5G Systems', in *IEEE Global Comm. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–7.
- [15] M. Jiang, M. Condoluci and T. Mahmoodi, 'Network slicing in 5G: An auction-based model', in *IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.
- [16] B. Chatras, U. S. T. Kwong and N. Bihannic, 'NFV enabling network slicing for 5G', in *2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN)*, Mar. 2017, pp. 219–225.
- [17] V. K. Choyi, A. Abdel-Hamid, Y. Shah, S. Ferdi and A. Brusilovsky, 'Network slice selection, assignment and routing within 5g networks', in *IEEE Conference on Standards for Communications and Networking (CSCN)*, Oct. 2016, pp. 1–7.
- [18] A. Mayoral *et al.*, 'Cascading of tenant SDN and cloud controllers for 5G network slicing using transport API and openstack API', in *Optical Fiber Comm. Conference and Exhibition (OFC)*, Mar. 2017, pp. 1–3.
- [19] T. Taleb *et al.*, 'On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration', *IEEE Comm. Surveys Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.
- [20] A. Lertsinsruttavee, A. Ali, C. Molina-Jimenez, A. Sathiseelan and J. Crowcroft, 'PiCasso: A Lightweight Edge Computing Platform', in *2017 IEEE 6th International Conference on Cloud Networking (CloudNet)*, Sep. 2017, pp. 1–7.
- [21] K. Kaur *et al.*, 'Edge Computing in the Industrial Internet of Things Environment: Software-Defined-Networks-Based Edge-Cloud Interplay', *IEEE Comm. Magazine*, vol. 56, no. 2, pp. 44–51, Feb. 2018.
- [22] J. Pan and J. McElhannon, 'Future Edge Cloud and Edge Computing for Internet of Things Applications', *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 439–449, Feb. 2018.
- [23] P. J. Braun, S. Pandi, R. Schmolle and F. H. P. Fitzek, 'On the Study and Deployment of Mobile Edge Cloud for Tactile Internet using a 5G Gaming Application', in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan. 2017, pp. 154–159.
- [24] *IEC 61850: Communication Networks and Systems for Power Utility Automation*, International Electrotechnical Commission TC57.
- [25] G. Karagiannis *et al.*, 'Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions', *IEEE Comm. Surveys Tutorials*, vol. 13, no. 4, pp. 584–616, 2011.
- [26] *OpenFlow Switch Specification Version 1.3.0*, Open Networking Foundation, 2018. [Online]. Available: [www.opennetworking.org](http://www.opennetworking.org).
- [27] *Floodlight Controller Version 1.2*, Project Floodlight, 2018. [Online]. Available: [www.projectfloodlight.org/floodlight/](http://www.projectfloodlight.org/floodlight/).
- [28] F. Kurtz, C. Bektas, N. Dorsch and C. Wietfeld, 'Network Slicing for Critical Communications in Shared 5G Infrastructures - An Empirical Evaluation', in *4th IEEE International Conference on Network Softwarization (NetSoft 2018)*, Canada, Jun. 2018.
- [29] R. El Hattachi and J. Erfanian, 'NGMN 5G Initiative White Paper', NGMN (Next Generation Mobile Network) Alliance, White Paper, 2015. [Online]. Available: [https://www.ngmn.org/uploads/media/NGMN\\_5G\\_White\\_Paper\\_V1\\_0.pdf](https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf).
- [30] *Ryu Software-Defined Networking Framework*, 2018. [Online]. Available: <https://osrg.github.io/ryu/>.
- [31] F. Kurtz, N. Dorsch and C. Wietfeld, 'Empirical Comparison of Virtualised and Bare-Metal Switching for SDN-based 5G Communication in Critical Infrastructures', in *IEEE Conference on Network Softwarization*, Jun. 2016.
- [32] *Iperf Community Edition*, Feb. 2018. [Online]. Available: <https://sourceforge.net/projects/iperf2/>.
- [33] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems 1588–2008*, The Institute of Electrical and Electronics Engineers, Inc.