# Synchronized Measurement Concept for Failure Handling in Software-Defined Smart Grid Communications

Fabian Kurtz, Nils Dorsch, Caner Bektas, Christian Wietfeld
Communication Networks Institute
TU Dortmund University
{fabian.kurtz, nils.dorsch, caner.bektas, christian.wietfeld}@tu-dortmund.de

***Abstract*—Future power grids - due to the ongoing development towards distributed, renewable generation - depend heavily on reliable real-time monitoring and control. Thus appropriate Information and Communication Technologies are crucial to guarantee stable grid operation. Software-Defined Networking is being considered for handling the diverse requirements of Smart Grid communications. In particular, simultaneous failures of power and communication systems need to be addressed nearly instantaneously (sub 10 milliseconds) to secure timely operation of critical protection functions. To evaluate such scenarios the exact time and sequence of events, in case of communication failures, need to be determined. Hence, a Smart Grid Software-Defined Networking testbed is introduced. It comprises a purpose-built hardware platform that synchronizes networking devices, allowing for highly accurate delay measurements. The platform also enables the precise interruption of communication links. It mimics the characteristics of physical link disruptions, facilitating the study of triggered effects and their time of occurrence. The resulting measurement set-up enables a conclusive comparison of different recovery approaches based on Software-Defined Networking. Finally the need for advanced failure detection mechanisms, such as Bidirectional Forwarding Detection or controller heartbeats, is shown.**

## I. INTRODUCTION

Both, transmission and distribution power systems, face massive alteration, sparked by the shift towards renewable energy generation. Distributed energy resources (DER) involve generation on different levels of the power grid, resulting in bidirectional power flows. Moreover, due to factors such as changing weather conditions and time of day, renewable energy generation is characterized by unsteady feed-in, demanding new measures for balancing generation and consumption. Finally, renewable energy sources such as off-shore wind parks are mainly located far from major hubs of consumption. This requires long distance power transmission, evoking additional stress on the power lines. Adaptive charging and discharging of Electric Vehicles (EV) provides means of compensating for varying power demands. Yet, all of these developments entail the need for precise monitoring and control of the power system to guarantee stable operation.

Hence, an appropriate Information and Communication Technology (ICT) infrastructure is required, ensuring reliable transmission of measurement values, statuses and commands in real-time [1]. We propose Software-Defined Networking (SDN), as a comprehensive solution for addressing this challenge. Building on the separation of data and control plane, SDN provides measures for flexible and dynamic network configuration, enabling integration of multiple approaches.

In this paper, we focus on the aspect of reliability, analyzing strategies for fast link failure detection and recovery. To ensure stable operation of the power grid, immediate handling of communication network faults is essential, especially if coinciding with electrical outages. Besides the actual design of recovery strategies, it is of foremost importance to provide means for measuring and verifying such strategies. We introduce a holistic low-cost measurement concept for inducing ICT infrastructure failures and quantify delays in the chain of events. To achieve this our Synchronized Link Interruption and Corruption Equipment (SLICE) utilizes Global Navigation Satellite Systems (GNSS) and Precision Time Protocol (PTP) [2] based synchronization. Standard failure detection mechanisms, using Ethernet link pulse, are shown to be insufficient for obtaining demanded end-to-end latencies below 10 milliseconds [3]. To meet this criterion, as defined by the IEC 61850 standard, additional detection mechanisms are needed. Bidirectional Forwarding Detection (BFD) or controller driven heartbeats (HB) [4] are candidates for this task. Their evaluation is facilitated by SLICE and performed on basis of a reference Smart Grid substation automation scenario with a ring-topology network, as given by Figure 1.

The structure of this paper is as follows: Section II reviews related work on Software-Defined Smart Grid communication networks as well as on fast failover. This is followed by an

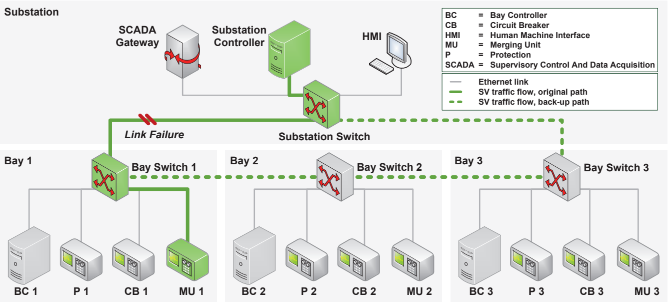

Figure 1: Substation Automation Scenario - IEC 61850 Communication Link Failure and Recovery

overview of the specifics of our SDN for Smart Grids concepts, including a description of mechanisms for failure detection and recovery (Section III). In Section IV we introduce the concept of SLICE. Section V comprises a description of the studied IEC 61850 substation automation scenario, while measurement results are presented and discussed in Section VI. Finally, Section VII gives a conclusion and an outlook on future work.

## II. RELATED WORK

The main contributions of this paper can be categorized into two domains: firstly SDN for Smart Grid communications and secondly measures for detecting and mitigating link failures. Sydney et al. were the first to propose SDN for the use in power system communications, comparing OpenFlow-based Quality-of-Service (QoS) with Multiprotocol Label Switching (MPLS) [5]. In [6] a substation automation system is proposed, which is configured on basis of SDN. The forwarding performance of SDN baremetal and software switches for the application in critical infrastructures is compared in [7]. Different strategies for failure detection and recovery in software-defined Smart Grid communication networks were compared and analyzed in [4]. Prior to its integration into OpenFlow, van Adrichem et al. proposed the application of BFD for fast failure detection in SDN, however not in the context of power system communciations [8]. In [9] backup wireless connections were employed for SDN-orchestrated link failure recovery in Smart Grid distribution networks. Fault-tolerance of multicast transmission in SDN-based Smart Grid communications is analyzed in [10]. An alternative to SDN-based fault tolerance is the Parallel Redundancy Protocol (PRP) [11] standard. It enables communication failover in substations via two separate networks. The iPRPR [12] protocol extends this principle to wide area applications in IP networks. Unlike SDN both require fully redundant networks, thus incurring higher costs, while iPRP does not provide provisions for QoS. In contrast to previous studies, this paper quantifies the delays of all events during failure and recovery process.

## III. SOFTWARE-DEFINED NETWORKING BASED FAILURE DETECTION AND RECOVERY

This section comprises a short review of our SDN for Smart Grids concept, followed by a survey of the link failure detection and recovery strategies compared in this work.

### A. *SDN for Smart Grids*

SDN builds upon the principle of splitting data and control functionalities to different planes [13]. Network control tasks are extracted from those devices within the network that physically handle information (e.g. routers, switches) and are concentrated on a dedicated, centralized instance. This entity, identified as SDN controller, thus gains a global view of the underlying infrastructure, to which it connects via the so-called southbound interface. One of the most common protocols for this purpose is OpenFlow (OF) [14]. As SDN decouples the network's intelligence from the physical layer, it is possible to deploy new algorithms on the controller without the need to upgrade devices in the field. Further, modern cloud computing platforms enable dynamic scaling of SDN controller resources. This enables new, increasingly complex and powerful network control algorithms, fully leveraging knowledge of traffic flow interdependencies. In contrast traditional networks are limited by fixed computing power at the distributed devices and thus are mostly restricted to algorithms developed prior to deployment. SDN, with its dynamic and flexible configuration, is an ideal basis for reliable, real-time Smart Grid communications. Previous work [15] introduced our Software-Defined Universal Controller for Communications in Essential Systems (SUCCESS). Originally forked from the Java-based Floodlight controller [16], it is specifically tailored for the communication demands of critical infrastructures. Therefore enhancements include mechanisms for fast failover, data traffic prioritization and queue configuration. Further, the controller supports dynamic adaptation of the communication network according to the needs of Smart Grid applications. This functionality can be accessed through Representational State Transfer (REST) messages, sent via the so-called northbound interface, requesting e.g. preferential transport of crucial switching commands. Other features of SUCCESS are multicast traffic handling and network state monitoring (both measurement-based and analytic). As a single controller reduces resiliency compared to distributed technologies like MPLS, a variety of mitigation strategies exists [17]. In this work, we capitalize on approaches of fast link failure detection and recovery. Here, the exact sequence of events and associated delays are reviewed as a crucial metric for time-critical Smart Grid applications.

### B. *Failure Detection and Recovery Mechanisms*

Handling of communication link failures can be divided into different tasks. First, the failure needs to be detected. This is particularly relevant for Ethernet-based infrastructures, due to their inherently slow detection mechanisms. Second, a strategy for resolving the failure is to be applied. Both steps might be addressed either in a centralized or decentralized manner.

*1) Detection Strategies:* Bidirectional Forwarding Detection (BFD) is a common approach for decentralized failure detection, frequently applied along with e.g. MPLS. As the name implies, BFD is capable of discovering link failures of the transmission and reception channels. To achieve this, connected network devices exchange lightweight Ethernet packets. The interval between these messages is determined by the Inter Transmission Time (ITT). As one lost packet does not necessarily denote a failed link, a detect multiplier is used. This way only several lost messages are recognized as a fault, of which the controller is then notified. SUCCESS enables remote configuration and activation of BFD, allowing the specification of parameters such as ITT and detect multiplier. Additionally, a similar, controller-based detection mechanism, the so-called Hearbeat (HB), is implemented. There lightweight Ethernet packets are encapsulated into OF PacketOut messages and send to the switches via the control network. These packets are decapsulated and forwarded as raw Ethernet packets over the data network. At the other end of the link the receiving switch re-encapsulates and sends the packets back to the controller as OF PacketIn messages. Again, if these packets are not received by the controller within a defined time interval, the link is declared failed. In contrast to BFD separate notifications are not needed as the SDN controller detects failures by itself. However, the HB creates additional control network and controller load. The latter is caused by periodic transmission and reception of HB messages with an ITT of a few milliseconds on all monitored network links. To limit controller load it is thus expedient to restrict the use of HB to critical links. Both approaches consume bandwidth of the data network, which is however less than $0.06\,\%$ (BFD) respectively $0.02\,\%$ (HB) of a Gigabit link.

*2) Recovery Strategies:* As for failure recovery, OF Fast Failover Groups (FFG) complement BFD ideally. Using FFG the controller pre-computes alternative paths for each possible link failure and pre-installs corresponding rules at the switches. These rules direct packets matching specified header criteria to a defined group of output ports. If the first port of the group is available, the traffic is forwarded to this port. Should the port fail, packets are switched to the next port available. This repair is performed locally, hence the controller needs to be notified for its view of the network status to remain valid. Additionally, the use of pre-computed failover paths might lead to non-optimal traffic routes but reduces reaction time, as shown in [4]. Hence, the use of FFG might necessitate subsequent post optimization of routes. In contrast, a controller-driven recovery strategy is employed in combination with HB. To achieve optimal alternatives, the recovery path is calculated on-demand, just after the HB timer expires. Thus, routing path calculations account for the current network state, resulting in a delay / load optimal solution.

## IV. Measurement Concept and Implementation for the Application in Smart Grids

Smart Grid applications specify exact requirements in terms of communication network performance. Low end-to-end latency, particularly in the presence of failures, is of great significance. The assessment of failure detection and recovery methods, thus calls for a stringent evaluation methodology. Performance differences and the chain of events are on a scale of single milliseconds. This can only be observed reliably with highly precise time measurements. Events such as failures and sending OF messages however, involve multiple devices distributed throughout the communication network, each with their own clock. Therefore a concept for synchronizing clocks and reducing their drift against each other is essential. Another crucial aspect is the creation of physical link failures, as they trigger the sequence of events under study. Section VI reveals that disabling communication links via software commands does not yield the same effects as physical failures. Damaging network cabling is inefficient, given that disconnecting links has essentially the same characteristics. Still, both approaches do not yield the exact time of when the connection is severed. Hence, a device which non-destructively induces link failures and records their timing, is necessary. The next section thus introduces the protocols employed in gaining and distributing a stable clock. Afterwards SLICE, which implements the functionalities described above, is presented.

### A. Synchronization of Distributed Network Devices

Synchronizing distributed devices requires a reference clock. It should be as precise as possible, since its precision and stability (i.e. lack of jitter) defines the boundary for all clocks derived from it. Atomic clocks meet these criteria but are expensive to acquire and maintain. A more cost effective solution is the use of time signals send by Global Navigation Satellite Systems (GNSS), such as the Global Positioning System (GPS). These Satellites include atomic clocks, providing time with nanosecond accuracy [18]. Once this reference source is obtained via a GNSS receiver it can be used to adjust a local Network Time Protocol (NTP) server. In NTP nomenclature GNSS reference clocks are called stratum 0 sources, while the local server itself resides on stratum 1, due to its reduced precision. The clocks of our distributed network devices (stratum 2) need to be coupled tightly to the NTP server. For this purpose we utilize the Precision Time Protocol (PTP), defined by IEEE 1588-2008, which is specifically designed for such use cases. PTP uses a master/slave design for distributing timestamps, providing sub-microsecond accuracy [18]. The precision of this setup also depends on protocol implementation and utilized devices. Hence, the next section describes the development of our SLICE platform, which acquires GNSS signals, hosts an NTP server and synchronizes our network devices.

### B. Synchronized Link Interruption and Corruption Equipment

SLICE, as indicated by its name, has two main functionalities. Communication network synchronization and link interruption. A Raspberry Pi 3 with the Broadcom BCM2837 System on Chip (SOC), running the Raspbian (Jessie, Kernel v4.4.13-v7+) Operating System (OS) serves as hardware platform. Through a General Purpose Input and Output (GPIO) interface it connectes to a GlobalTop GPS module (model FG-PMMOPA6H) equipped with the MediaTek MT3339 chipset. This sends a Pulse Per Second (PPS) signal through the GPIO pins with nanosecond precision [18]. The NTP server (NTPd v4.2.8p7) is coupled to the GPS clock and distributes time throughout the network via PTP (PTPd 2.3.1).

SLICE's other task is the precise interruption of links on the physical layer. To avoid the introduction of forwarding delays, we devised the circuit design given by Figure 2. One copper wire of an Ethernet cable is soldered to the emitter of a transistor (BC547C), with the collector connected to ground. The transistor's base is bonded to a GPIO pin, whose $3.3\,\mathrm{V}$

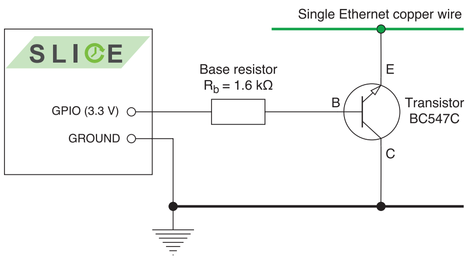

Figure 2: Circuit Layout for Interruption of Copper-based Ethernet Links, used for SLICE

are reduced to an appropriate voltage via a $1.6\,\mathrm{k\Omega}$ resistor. Thereby the transistor can be switched on, pulling the wire to ground potential, disrupting communication. This circuit is installed to all eight wires connecting two female Ethernet plugs. Hence, SLICE not only reproduces link failures reliably, but can be integrated into any path by attaching cables to its plugs. As our device also acts as a highly precise clock, the exact time of link failures is recorded.

## V. IEC61850 SUBSTATION AUTOMATION SCENARIO AND TEST SETUP

This section provides a detailed description of the substation scenario under study and the corresponding testbed set-up.

### A. Scenario Description

The International Electrotechnical Commission (IEC) defines in its standard 61850 (IEC 61850) a comprehensive set of rules for substation automation, spanning from detailed data models to messaging services. In recent years, the standard spread to all kinds of power system applications, such as the connection of EV or DER. Aiming to enhance interoperability and reducing investment costs, IEC 61850 promotes the use of Ethernet equipment within substation environments, replacing proprietary, legacy connections of every device. Subsequently, fast recovery from communication link failures is a major concern for reliable operation of future substations.

The most important communication protocols, considered in IEC 61850, are Sampled Values (SV), Generic Object Oriented Substation Events (GOOSE) and Manufacturing Message Specification (MMS). While the latter uses TCP/IP-based client-server communication (for e.g. configuration, measurement reports), the former two protocols apply Ethernet messages directly. SV transports measurement values in fixed time intervals, usually every $250\,\mu$s, from Merging Units (MU) to bay or substation controllers as well as to protection devices. GOOSE messages however are employed for status updates and signaling commands, e.g. triggering circuit breakers.

For evaluating the measurement concept, described in Section IV the SDN testbed has been set up as an IEC 61850 substation in ring topology with three identical bays, as shown in Figure 1. Each merging unit sends out SV messages to its respective protection device and bay controller. Also, measurement values are relayed to the substation controller, located on the substation level along with the Supervisory Control and Data Acquisition (SCADA) gateway and the Human Machine Interfaces (HMI). Besides, GOOSE messages are exchanged between circuit breakers, protection devices, bay controllers and the substation controller in intervals of $1\,$s during normal operation. In case of an emergency the ITT is decreased drastically to milliseconds.

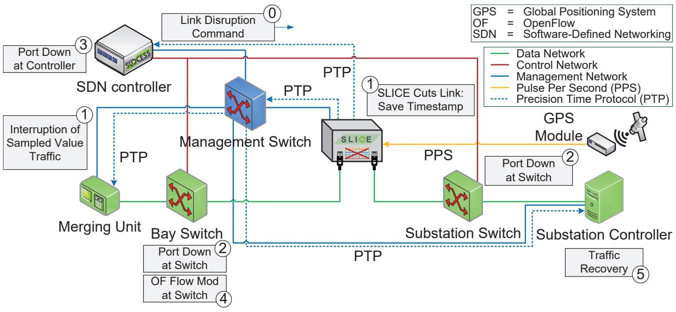

Figure 3: Measurement Setup for Communication Link Disruption, including SDN Testbed and SLICE

### B. Measurement Setup

Figure 3 presents a relevant subset of the laboratory setup created for the purposes of this paper. Measurements devices are depicted, as well as the pertinent sections of the scenario given in Figure 1 (c.f. matching designations and colour coding). SUCCESS interfaces with both switches via a separate, out-of-band SDN control network (red). It also connects to the management network (blue), which enables remote access to itself, the merging unit and the substation controller, without interfering with measurements. The data network (green), linking grid devices via bay and substation switch, corresponds to the path outlined in Figure 1. Here, Smart Grid traffic flows are handled, making it the focal point of our study. Network separation obviates any interference between Smart Grid communication and measurement infrastructure. SLICE is located in the Figure's center. A GPS module (yellow connection) provides the reference clock signal, which is distributed via the management network utilizing PTP. SLICE connects bay and substation switches physically, i.e. free of forwarding delay. In response to failing this path, the configured detection and recovery strategies are triggered.

The entire measurement cycle unfolds as follows. It should be noted, that the numbering in Figure 3 is according to the sequence of events in case of controller failover without further detection mechanisms. First the SDN controller sends a link disruption command (0) to SLICE. This device then severs the connection between bay and substation switch and records the corresponding timestamp (1). Next the switches observe a port down event (2), after which the SDN controller is notified (3). Updated rules (i.e. OF FlowMods), on how to forward messages involving the failed link, are created by the controller and arrive at the affected switches (4). Finally, packets from the merging unit arrive at the substation controller (5) via the newly created path. Traffic recovery is thus completed.

The SDN testbed includes the following components. Substation and bay switches are created by running Open vSwitch (v2.5.2) on four Intel Xeon D-1518 servers. Each of these virtual switches is equipped with a two Port I2170-LM and two four Port I350 Intel 1GBase-T Ethernet Network Interface Cards (NIC). Remaining Ethernet interfaces connect to control, respectively management networks. Ubuntu Server 16.04.2 LTS (v4.4.0-77-generic x86-64 Kernel) is used as OS. OS and hardware of our SUCCESS platform are identical to those of the virtual switches. However, it does not connect to the data network. Merging unit and substation controller are Intel Celeron J1900 based, with a single two Port I2170-LM NIC. A Zyxel GS1900-24E switch, with forwarding delay negligible for the purposes of this paper [7], handles management and PTP synchronization traffic.

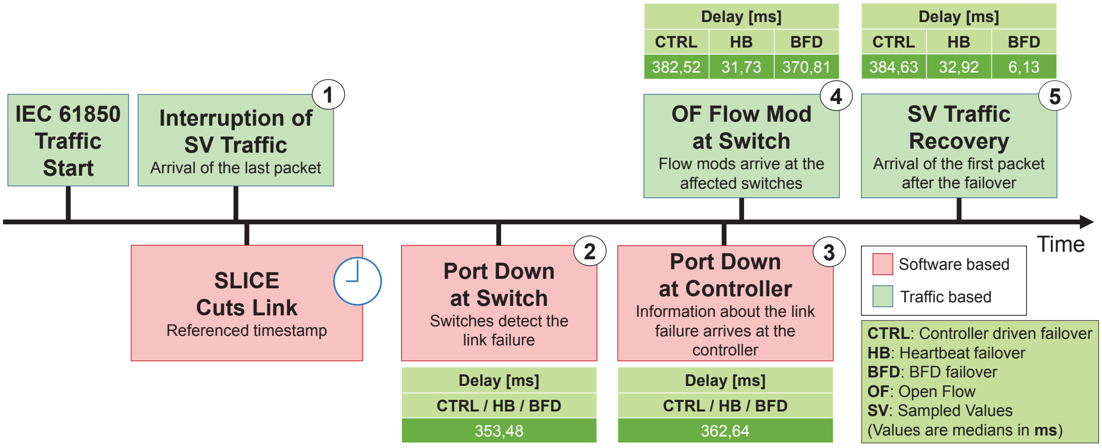

Figure 4: Chain of events during the fast failure detection and recovery process (graphical distances not according to scale)

## VI. EVALUATION RESULTS

The goal of this evaluation is a) to compare the performance of different failure detection and recovery strategies and b) to quantify the delay of each step during the recovery process. In previous work [15], we demonstrated significant deviations between physical link disruptions and deactivating ports via software. If ports are set inactive within a device, the exact timing is registered, yet detection delays are neglected. Using SLICE, precisely timed link interruption is achieved, while obtaining the exact characteristics of physical link disruption. To study delays induced by different failover approaches in detail, SLICE is attached to the link connecting substation and bay switch one (c.f. Figure 1). As SLICE disrupts the link, all transmissions between bay one and the substation controller are halted. Communication is restored as SDN re-directs traffic to the back-up path via bay switches two and three. In the following, the time of the link disruption is used as reference for all delay computations. Figure 4 illustrates the failover process in-depth, beginning from the start of IEC 61850 traffic. The numbering of each step, is in alignment with Figure 3. Traffic interruption (1) is marked by the last packet transmitted successfully, thus beginning shortly before the actual link disruption. Next, the interval between disconnection and its registration at the affected switch is determined using Ethernet link pulse, yielding a median of 353.48 ms (2). Deviations from this median as well as the corresponding distribution of delays can be obtained from the leftmost violin plot in Figures 5 - 7. Figure 5 shows failure detection and recovery delays without any advanced detection mechanisms. HB and BFD delays are illustrated in Figures 6 and 7. The figures also show time intervals between the different events plotted, such as control and data network delay.

Subsequent to failure registration at the switch, the SDN controller is notified (3), which incurs a median delay of 362.64 ms. It has to be noted, that - in every case studied - this delay refers to the reception of the regular OF PortDown message. Switches issue this notice in response to a link failure detected through the absence of Ethernet link pulses. Accordingly, the delay is identical in all three cases. Following the default case's chain of events, the controller processes the OF PortDown message, computes alternative routes and signals these to the switches via OF FlowMod packets. Reception of these OF FlowMods requires a median latency of 382.52 ms (4). In contrast, if HB-based failure detection is employed, OF FlowMods are generated in response to a HB time-out, reducing delay (4) drastically to a median of 31.73 ms (c.f. Figure 6). In case of BFD, OF FlowMod reception delay remains in the range of more than 300 ms. However, in this instance recovery measures are already in effect. This interval is thus only relevant for post optimization mechanisms, which transfer recovered traffic streams onto load or delay optimized paths. Finally, the standard controller-driven case achieves full traffic recovery after 384.63 ms (5). It is therefore not suitable for use in substation automation. HB reduces the median duration for recovery to 32.92 ms (HB ITT 3 ms, time-out 20 ms), offering carrier-grade performance [4] but falling short of Smart Grid targets. Figure 7 shows further improvements, achieving a median delay of 6.13 ms, when using BFD with an ITT of 1 ms and a detection multiplier of 5. End-to-End (E2E) latency, which has to be met despite failures, is thus below the 10 ms threshold defined by IEC 61850. Yet, as BFD employs pre-calculated paths, subsequent optimization is expedient.

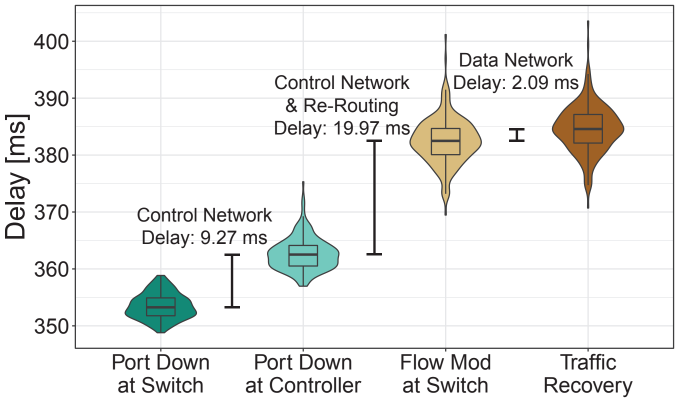

Figure 5: Controller-driven IEC 61850 traffic recovery delays

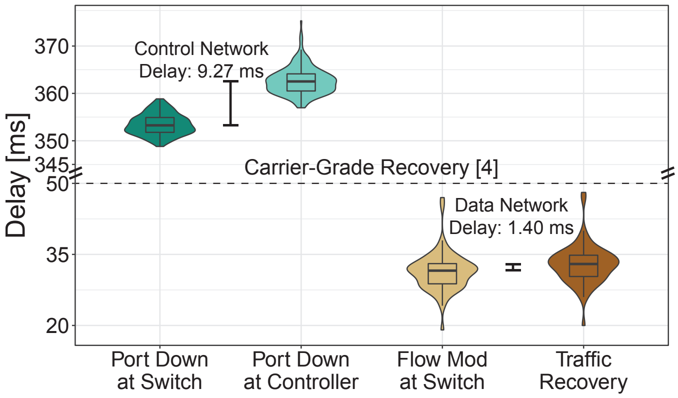

Figure 6: Heartbeat-driven IEC 61850 traffic recovery delays

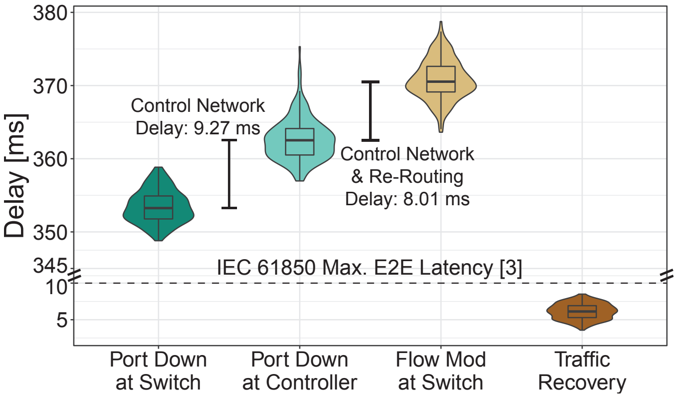

Figure 7: BFD-driven IEC 61850 traffic recovery delays

## VII. Conclusion and Outlook

This work conducts a detailed analysis of communication link failures in Smart Grids, obtaining a precise view of the whole process from interruption to recovery. To achieve this, SLICE is introduced, implemented and evaluated as a dedicated, cost-efficient device for link interruption and network synchronization. Build on commonly available hardware it enables the user to create physical connection failures, while obtaining full control of the whole interruption process. To enable fast failover, strategies for fault detection and recovery are introduced on basis of our SUCCESS SDN controller platform, tailored specifically to the needs of Smart Grid communications. A realistic IEC 61850 substation automation scenario, implemented in our SDN testbed in conjunction with SLICE, is presented. Subsequently, different approaches for failure detection and recovery are compared. Additional detection mechanisms are shown to reduce failover duration by up to 98 %, thus meeting Smart Grid requirements. Applying our advanced measurement concept, we provide exact timing of all events during the failure and recovery process.
Future work will focus on individual packet error generation, countermeasure design and improved precision via PTP switches. Moreover, an enhanced revision of SLICE is aimed at connecting to and interrupting multiple links.

## Acknowledgement

This work has been carried out in the course of research unit 1511 *'Protection and control systems for reliable and secure operations of electrical transmission systems'*, funded by the German Research Foundation (DFG) and the Franco-German Project *BERCOM* (FKZ: 13N13741) co-funded by the German Federal Ministry of Education and Research (BMBF). Also, we would like to thank O. Belov, L. Koblenzer, I. Laukhin and N. Smolnikow for their valuable input.

## References

[1] Y. Yan, Y. Qian, H. Sharif and D. Tipper, 'A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges', *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 5–20, 2013.

[2] *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems 1588–2002*, The Institute of Electrical and Electronics Engineers, Inc.

[3] *IEC 61850: Communication Networks and Systems for Power Utility Automation*, International Electrotechnical Commission TC57.

[4] N. Dorsch, F. Kurtz, F. Girke and C. Wietfeld, 'Enhanced Fast Failover for Software-Defined Smart Grid Communication Networks', in *IEEE Global Comm. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[5] A. Sydney, J. Nutaro, C. Scoglio, D. Gruenbacher and N. Schulz, 'Simulative Comparison of Multiprotocol Label Switching and OpenFlow Network Technologies for Transmission Operations', *IEEE Transactions on Smart Grids*, vol. 4, no. 2, pp. 763–770, 2013.

[6] A. Cahn, J. Hoyos, M. Hulse and E. Keller, 'Software-Defined Energy Communication Networks: From Substation Automation to Future Smart Grids', in *IEEE International Conference on Smart Grid Communications*, 2013, pp. 558–563.

[7] 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.

[8] N. van Adrichem, B. Van Asten and F. Kuipers, 'Fast Recovery in Software-Defined Networks', in *European Workshop on Software Defined Networks (EWSDN)*, Sep. 2014, pp. 61–66.

[9] A. Aydeger, K. Akkaya, M. H. Cintuglu, A. S. Uluagac and O. Mohammed, 'Software defined networking for resilient communications in Smart Grid active distribution networks', in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–6. DOI: 10.1109/ICC.2016.7511049.

[10] T. Pfeiffenberger, J. L. Du, P. B. Arruda and A. Anzaloni, 'Reliable and flexible communications for power systems: Fault-tolerant multicast with SDN/OpenFlow', in *International Conference on New Technologies, Mobility and Security*, Jul. 2015, pp. 1–6.

[11] *IEC 62439-3: Industrial communication networks - High availability automation networks - Part 3: Parallel Redundancy Protocol (PRP) and High-availability Seamless Redundancy (HSR)*, International Electrotechnical Commission TC65.

[12] M. Popovic, M. Mohiuddin, D. C. Tomozei and J. Y. L. Boudec, 'iPRP - The Parallel Redundancy Protocol for IP Networks: Protocol Design and Operation', *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1842–1854, Oct. 2016.

[13] N. McKeown *et al.*, 'OpenFlow: Enabling Innovation in Campus Networks', *SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[14] *OpenFlow Switch Specification Version 1.3.0*, Open Networking Foundation, 2012. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.3.0.pdf.

[15] N. Dorsch, F. Kurtz, H. Georg, C. Hägerling and C. Wietfeld, 'Software-Defined Networking for Smart Grid Communications: Applications, Challenges and Advantages', in *IEEE International Conference on Smart Grid Communications*, Nov. 2014, pp. 422–427.

[16] *Floodlight Controller Version 1.2*, Project Floodlight, 2015. [Online]. Available: http://www.projectfloodlight.org/floodlight/.

[17] K. C. Fang, K. Wang and J. H. Wang, 'A fast and load-aware controller failover mechanism for software-defined networks', in *International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Jul. 2016, pp. 1–6.

[18] H. Guo and P. Crossley, 'Design of a Time Synchronization System based on GPS and IEEE 1588 for Transmission Substations', *IEEE Transactions on Power Delivery*, no. 99, 2016.