Published in: SmartGridComm 2014 - 2014 IEEE International Conference on Smart Grid Communications
DOI: 10.1109/SmartGridComm.2014.7007683

Software-Defined Networking for Smart Grid Communications: Applications, Challenges and Advantages

Nils Dorsch, Fabian Kurtz, Hanno Georg, Christian Hägerling and Christian Wietfeld
TU Dortmund University, Germany
Communication Networks Institute (CNI)
Email: {nils.dorsch, fabian.kurtz, hanno.georg, christian.haegerling, christian.wietfeld}@tu-dortmund.de

***Abstract*—Future power systems are characterized by a high degree of complexity with a large number of intelligent devices, exchanging and processing both huge amounts of data and real-time critical information. Accordingly reliable, real-time capable and secure communication networks are required for enabling autonomous monitoring, management and control to guarantee stable power system operation. In this paper, we present and analyse a flexible and dynamic network control approach based on Software-Defined Networking (SDN) for meeting the specific communication requirements of both distribution and transmission power grid. Therefore a testbed is introduced, enabling the evaluation of multiple failure scenarios such as link disturbance and congestion by analysing corresponding fast recovery and prioritization solutions. The performance and robustness of the developed strategies is shown using highly-critical monitoring and control messages on basis of IEC 61850 and considering the mutual impact with low priority background traffic. Results indicate the advantages of SDN compared to traditional routing and Quality-of-Service mechanisms, providing a more reliable communication network, which is able to handle complex failure scenarios. In particular, SDN enables the integration of diverse network management functions and thus offers the power system new options for dealing with faults even in the case of overall outages. On the basis of these results, we demonstrate challenges and derive future benefits for a SDN-enabled Smart Grid communication network, holding the potential to evolve into a self-healing infrastructure.**

I. Introduction

Currently, the energy sector is undergoing massive changes, evolving the traditional grid into a Smart Grid. In particular, Smart Grids create heightened demand for communication, so that *Information and Communication Technology* (ICT) infrastructures play a vital role in future power systems. Among the new challenges for the power system, the most important ones are integrating renewable *Distributed Energy Resource*s (DERs), successfully applying *Demand Side Management* (DSM) and dealing with increased energy trade [1]. Furthermore developments like *Automated Meter Reading* (AMR) and *Customer Energy Management Systems* (CEMS) lead to the inclusion of potentially millions of consumers and prosumers into the overall Smart Grid architecture. The proliferation of Hybrid and *Electrical Vehicles* (EVs) is another aspect that leads to rising requirements for coordination and therefore additional control communication. All these factors exert additional stress on the power system and might even compromise its stability.
Thus, in order to guarantee safe and stable operation, it becomes essential to closely monitor the power system and intervene more frequently to countervail imbalances. Subsequently, rising numbers of *Intelligent Electronic Devices* (IED) will have to be installed within the grid, which enable processing and exchanging monitoring and control data. This leads to larger overall data volumes and a high degree of complexity which needs to be handled by the ICT infrastructure [2]. Therefore, the development towards Smart Grids requires high performance communication networks, which enable reliable, robust, time-critical and secure data transmission. One means to achieve such a communication infrastructure could be the application of *Software-Defined Networking* (SDN), which we analyse in this paper. Subsequently, an introduction to the concept of SDN as well as related work is provided in Section II. Section III details the Use Cases serving as a basis for this article. The test-setup and results are laid out in Section IV. Finally Section V provides a conclusion and an outlook on future work.

II. Software-Defined Networking - Approach and Applications

This section sums up the SDN concept with respect to its application in the Smart Grid context and outlines related work.

A. Concept of Software-Defined Networking

SDN is a new approach towards managing and controlling communication networks, which originally has been developed for experimenting with new algorithms and protocols on public communication networks [3]. It bases upon the idea of separating control and data plane as depicted in Figure 1, and applies a programmable controller instance - the SDN controller - for dynamically manipulating / influencing the behaviour of the communication network. Figure 1 has been extended to capture our view of Smart Grid integration of SDN.
As control logic is moved into the SDN controller, all network devices can be realised as simple switches, matching arriving packets against entries in their flow table and looking up corresponding actions such as dropping or forwarding. In case a packet does not match a known traffic flow, the packet is transmitted to the SDN controller, which decides on the actions for this new traffic flow and pushes instructions to the switches for establishing appropriate flow table entries. For the communication between the switches and the SDN controller via the Southbound API, the OpenFlow (OF) [4] protocol, standardized by the Open Networking Foundation, is applied (cf. Figure 1). On top of the controller, a variety of different Software-Defined Networking aware applications might be developed for influencing the behaviour of the communication network. Such applications could benefit from the information and flexibility that is available through the controller.

B. Related Work

During the past few years SDN became a major topic of interest for developing new communication concepts. Yet, to the

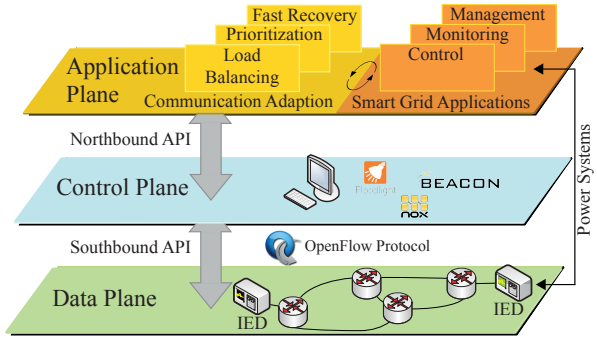

Fig. 1: Proposed Multi-layered SDN for Smart Grids

best of our knowledge, up to now there has been only limited effort to exploit the benefits of Software-Defined Networking for Smart Grid communications.
In [5] SDN was applied for developing self-configuring IEDs for substation automation, which were tested using emulation of a substation communication network. Multicast transmission of *Phasor Measurement Unit* (PMU) data was handled by a SDN enabled network in [6], adapting data rates to different receiver requirements. Sydney et al. [7] studied the application of SDN as an alternative to *Multiprotocol Label Switching* (MPLS) for wide area communication within the power system. It was shown that this allows for the integration of all relevant aspects of MPLS and provides reasonable data rates, while offering a large degree of flexibility and openness towards new concepts.
In comparison to these first approaches to SDN-based communication networks in Smart Grids, we focus on fault-tolerant, real-time capable and resource-efficient communication enabled by new algorithms on top of the SDN controller, validated with the help of our testbed setup. Moreover, we achieve a holistic view of the power system, comprising time-critical communication within the transmission power grid and Quality-of-Service (QoS) demands for communicating over heterogeneous infrastructures within the distribution power grid.
Comparable SDN-enabled approaches for fast recovery from link failures have been proposed in [8] and [9] in other contexts. Yet, especially for Smart Grids these applications are of major importance since robustness of the communication network is a key enabler for automated control. In contrast to previous approaches, the impact of realistic traffic patterns as well as physical link disconnections have been considered in this work.

III. Communication within power systems - Use Cases

Corresponding to the structure of Smart Grids, communication demands vary significantly depending on the specific application and its corresponding level of the power system. While the transmission power grid requires time-critical and reliable transfer of monitoring and control messages, heterogeneous technologies, protocols and applications pose a major challenge for communication within the distribution power grid. In recent years, IEC 61850 [10] has been established as the predominant standard for communication within power systems. Originally, it has been proposed by the *International Electrotechnical Commission* (IEC) *Technical Committee* (TC) 57 in 2002 for substation automation in order to achieve interoperability of devices and services. However, its scope has been enhanced by several extensions and technical reports, so that it also defines concepts for wide area communication, the connection of DERs and electric vehicles by now. Moreover, IEC 61850 comprises more than communication services only, specifying a complete data structure for describing devices and their attributes by using the *Substation Configuration Language* (SCL) as a configuration language. In regard to communication, it provides three major services: *Manufacturing Message Specification* (MMS), *Generic Object Oriented Substation Events* (GOOSE) and *Sampled Values* (SV). MMS is a client-server based protocol, using the TCP/IP stack, and can be used for multiple purposes such as communication on substation and wide area level. In contrast, GOOSE and SV are employed mainly for (multicast based) communication on substations' bay level, whereat data is encapsulated directly into Ethernet frames. While the GOOSE service is used for sending switching commands and providing status updates, SV messages contain measurement values.
In this work, IEC 61850 compliant communication has been applied, serving as basis for realistic traffic mapping for Smart Grid ICT infrastructures. Subsequently, we aim at achieving highly fault-tolerant and real-time capable communication networks, considering the specific requirements of IEC 61850 traffic. In the following the two Use Cases considered in the scope of this paper are presented.

A. Use Case A: Controlling the Transmission Power Grid

As for monitoring and control of the transmission power grid two domains of communication can be identified: intra-substation and wide-area communication. The latter covers data exchange between different substations as well as the control centre's *Supervisory Control and Data Acquisition* (SCADA) system. To ensure stable operation of the power system, its state has to be monitored in real-time. Beyond that, switching commands have to be transferred within a few milliseconds from protection devices and/or control units to circuit breakers within substations. Otherwise, stable operation of the transmission power grid might be disturbed. In the worst case, if additional failures occur simultaneously, fault recovery might not be possible, leading to cascading outages and blackouts [11]. Concerning reliability, the primary focus is to guarantee fast recovery of highly time critical communication services after link failures or errors in order to maintain grid stability. In addition, QoS mechanisms are introduced, ensuring prioritized handling of critical messages, such as switching commands, by establishing data rate, respectively link reservation.

B. Use Case B: Managing the Distribution Power Grid

Distribution grids and their associated devices employ a multitude of protocols for communication. IEC 61850 is used alongside IEC 60870, IEC 62056, IEC 61334 and others. However the most important distinction that has to be made in communications systems in the distribution domain compared to the transmission grid is the use of heterogeneous, shared resources. Especially AMR and CEMS enabled households may utilise public access networks that have not been designed with the exclusive use for Smart Grid communication in mind. Here, technologies like DSL, DOCSIS and cellular networks are pervasive and used for applications including Internet access and streaming services. One challenge that results from this is the need to reliably meet the requirements imposed by Smart Grid protocols in the presence of user-originated background traffic and failures. Therefore several QoS classes are introduced (cf. Table I), placing a higher priority on grid traffic, enforced by the SDN controller.

IV. Test Setup and Results of SDN-enabled Smart Grid Communication

Starting with the testbed setup this section reveals extended possibilities of SDN-based networks by means of two different scenarios. Following the scenario descriptions, algorithms for establishing the required capabilities at the SDN controller and analysis results are presented.

A. Testbed Setup

The SDN4SmartGrids testbed comprises four switches, one SDN-controller and two servers for generating and one client for receiving traffic as shown in Figure 2. As for the switches, four identical workstations have been used, running Open vSwitch 2.1.0 [4] as kernel module on basis of 64 Bit Ubuntu 13.04 Server (3.8.0-30-generic Kernel). Each switch is equipped with one integrated Intel I217-LM 1000 Base−T (IEEE 802.3ab) Ethernet Controller for connecting to the control network and one 4 Port Intel I350 1000 Base−T Ethernet Controller for operating within the data network. The SDN-controller has been developed by enhancing the OpenFlow Controller Beacon 1.0.4 [12] based on Java JDK 1.7 running on Windows 7 x64. Client and server workstations connect to the data network with onboard 1000 Base−T RTL8167 Realtek adapters, using Windows 7 x64 as OS, as well.
IEC 61850 compliant traffic has been generated applying the open-source library libIEC61850 [13], written in standard C, for MMS reports as well as a C implementation of the SV service for SV messages, developed at the Communication Networks Institute. For testing, both SV and MMS messages are sent in intervals of 1 ms, using packet size of 122 Byte and 684 Byte.

B. Scenario 1: Fast Recovery for Smart Grid Communications

This scenario deals with enabling fast recovery after disturbance of a communication link. Providing such functionality is of great importance for ensuring reliable operation of communication networks in critical environments such as substations of power systems. In particular, alternative routes through the network need to be established immediately, guaranteeing the transmission of monitoring and control traffic. Therefore a proactive algorithm for calculating alternative paths through the network has been developed and integrated into the SDN controller. The algorithm's performance has been assessed by measuring the duration of traffic interruption as well as processing times at the controller and switches.
First, a brief description of the algorithm, which is applied for dealing with communication link disconnection, is given:

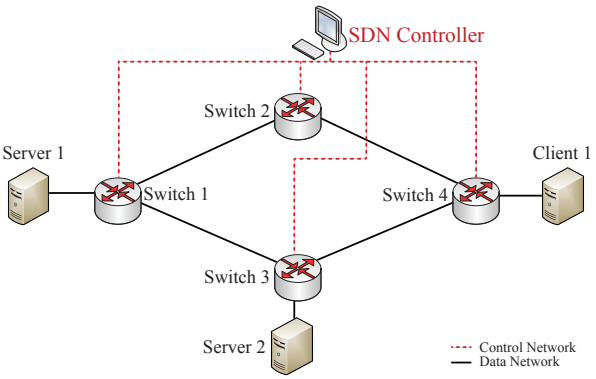

Fig. 2: Set-up of the SDN4SmartGrids Testbed with Data and Control Network

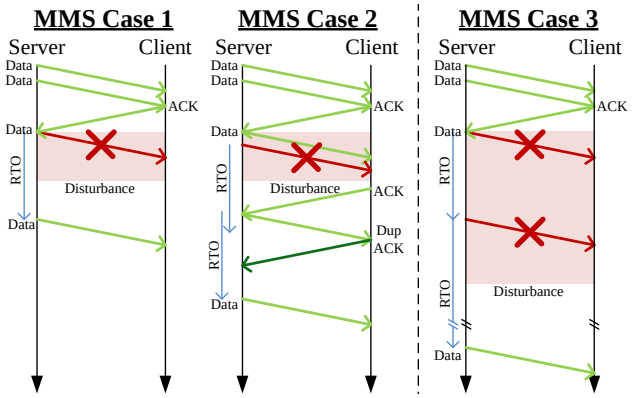

Fig. 3: MMS-TCP Flowchart Showing Different Recovery Cases

1) **Alternative shortest paths:** Alternatives are calculated for each pair of route and possible link failure, using alternative topologies which exclude the respective link.
2) **Mapping to switch configurations:** The alternative paths are converted into switch configurations, preparing corresponding OFFlowMod messages for adding, reconfiguring or deleting traffic flow entries at the switches.
3) **Monitoring of active flows**: The SDN controller keeps track of active traffic flows and their routes, considering OFFlowRemoved messages received from switches.
4) **Port status notification:** In case of a link failure, an OFPortStatus message is issued by the switches connected to the respective link and sent to the SDN controller.
5) **Re-routing:** Pre-calculated alternative routes are looked up for all affected traffic flows and corresponding OFFlowMods are sent out to the switches for re-routing.

Applying this algorithm, multiple measurements have been executed to verify and analyse its effect. The experimental design of this scenario can be conceived by means of Figure 2: MMS reports respectively SV messages, both containing measurement values, are transmitted from Server 1 to Client 1 using either the upper path (via Switch 2) or the lower path (via Switch 3). During transmission, one of the active communication links is disconnected by a) physical disconnection of an interface or b) by software command. Figure 4 (top) shows the overall recovery times of SV and MMS traffic, in terms of a cumulative distribution function, considering both cases. In case of SV messages and link disconnection by command, the mean down time of transmission amounts to 87.16 ms (median: 85.17 ms), whereas for physical link disconnection the mean delay increases to 360.64 ms (median: 358.80 ms). Hence, port status detection by the OS induces additional delay in the range of 210 to 305 ms. A more complex behaviour can be observed for TCP-based MMS reports due to reliability mechanisms, which apply acknowledgements (ACKs) and retransmissions. Accordingly, recovery time depends on the following TCP-specific parameters, which are set to Windows 7 default configuration: *Retransmission Time-Out* (RTO) (300 ms), acknowledgement frequency (2 packets) and delayed acknowledgement timer (50 ms). Therefore, Figure 3 distinguishes three different cases, which might occur when a link is disconnected during TCP based communication, explaining the effects encountered in Figure 4.
MMS Case 1: In Case 1, the link is disconnected before or during the transmission window's first packet transfer. The packet will not be received by the client and no ACK is issued, wherefore retransmission begins after the RTO elapses. This case applies to MMS reports with recovery times in the range of 300 ms after the link is disconnected by command.

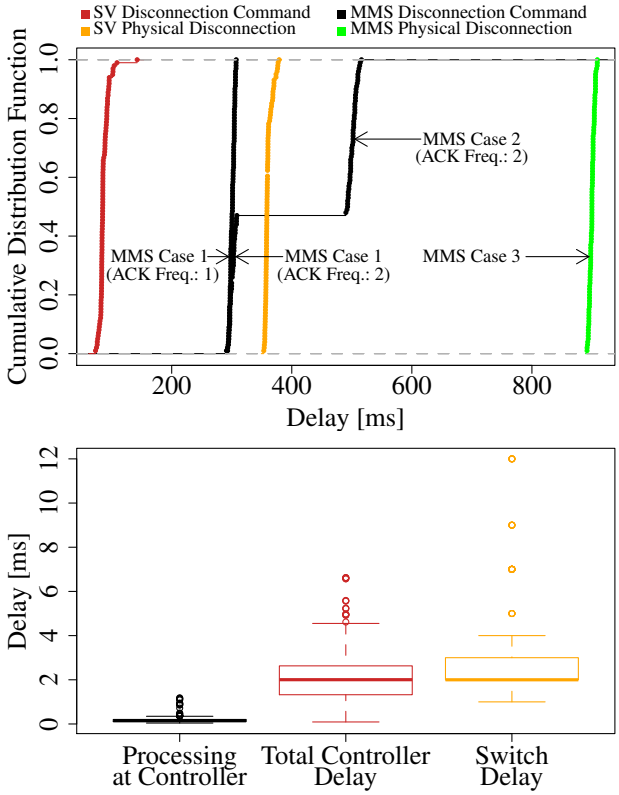

Fig. 4: Scenario 1: Failover Delays for SV and MMS Messages at Receiver (top) and Controller (bottom)

MMS Case 2: In contrast, the second case covers transmissions which have been interrupted after the successful transfer of at least one packet of the send window. In this case an ACK is issued for the last packet, since the client does not receive any further packets. This ACK reaches the server before the expiry of the RTO, causing the server to start sending the next transmission window. However, the new packet's sequence number does not match the packet order at client side, wherefore the client replies with a duplicated ACK. Thus, both server and client remain inactive until the expiry of this new packet's RTO, which results in additional delay until the transmission is actually recovered. In Figure 4 MMS reports complying with this case exhibit recovery times in the range of 500 ms, occurring after command-based link disconnection. Combining the measurements of Case 1 and 2 yields a mean recovery time of 407.21 ms for command-based link disconnection. Yet, the distribution among both cases might be varied by modifying the ACK frequency. A frequency of 1 leads to all packets being delayed by about 300 ms, whereas higher frequencies shift the major part of events to 500 ms.

MMS Case 3: The third case applies to all MMS packets after physical link disconnection, implying a mean down time of 899.99 ms. Here, the minimum time until a new valid route through the network is available - given by SV messages' recovery times - exceeds the RTO. Thus, retransmission is attempted - no matter at which time communication was interrupted - before recovery, resulting in duplication of the RTO and additional delay until the connection can be re-established.

Considering the described impact of the RTO, recovery delay might be improved by reducing the RTO with SV results illustrating the optimum that could be achieved.

Further insight to the recovery function is given by the box plots in Figure 4 (bottom), holding different delays measured during recovery at the SDN controller and the switch. The controller requires a mean processing time of 0.18 ms (median: 0.15 ms) for looking up alternative paths and issuing corresponding OFFlowMod messages, whereas the overall delay at the controller amounts to a mean value of 2.01 ms (median: 1.99 ms), considering the time between reception of the OFPortStatus message and the transmission of OFFlowMods. At the affected switches, on average a period of 2.61 ms (median: 2 ms) has been measured between the identification of the interface disconnection and the reception of new flow table entries from the SDN controller, yet in the worst-case this time prolonged to 12 ms.

Moreover, to cope with link errors disturbing transmissions, the previous algorithm might be extended as shown below. Here, port statistics requests are utilized by the SDN controller for identifying link errors, since automatic detection is impossible.

1) **Statistics Request:** The SDN controller sends OFPortStatisticsRequests periodically to all switches.
2) **Statistics Analysis and Re-routing:** Based on the statistics, the SDN controller determines whether the error bound for a link is exceeded and OFFlowMod messages for re-routing need to be sent.

Compared to routing protocols such as Open Shortest Path First (OSPF) our SDN-based approach enables faster recovery as it responds to network events. Applying this algorithm, the major part of the recovery time has to be attributed to a) internal processing at the switch previous to interface disconnection detection and b) characteristics of the communication protocols. In contrast, the recovery time in traditional communication networks depends on network discovery frequency, which is 10 s for OSPF, resulting in a mean delay of 5 s. Besides, also MPLS offers ways for fast recovery, using e.g. Resource Reservation Protocol Traffic Engineering (RSVP-TE) or Ethernet auto-negotiation. Failure detection in RSVP-TE bases on exchanging hello messages, implying trading fast detection (minimum 5 ms) for high traffic load on the data network. Using auto-negotiation in [14], on average 2.52 ms are required for failure notification and local repair, which is slightly faster than our solution (0.09 ms). Yet, we provide global recovery and analysed the incurred delay of traffic streams. Thus, in most cases relying on OSPF or MPLS would involve a trade-off between reducing recovery time by increasing the discovery frequency and ensuring little additional load on the data network, whereas in a SDN-enabled network both issues could be handled simultaneously.

C. *Scenario 2: Ensuring Smart Grid Quality-of-Service*

Scenario 2 covers the introduction of QoS mechanisms in order to achieve prioritized handling and improve the real-time properties of certain, critical traffic flows, e.g. switching commands. Here, QoS is realized by establishing queues at each port of every switch, which guarantee specified minimum and maximum data rates of traffic flows. Queue configuration is done directly via Open vSwitch, since up to now OpenFlow does not support the configuration of queues. Yet, the SDN controller is

TABLE I: QoS Classes for Smart Grid Communications [10]

Class	Service	Min Data Rate	Latency
5	Network Control	5 Mbps	$<$ 10 ms
4	Smart Grid Control	20 Mbps	10 – 100 ms
3	Smart Grid Monitoring	50 Mbps	10 – 100 ms
2	Real-Time Video, Voice Traffic	30 Mbps	4 s, 150 ms
1	Data Transfer	20 Mbps	-

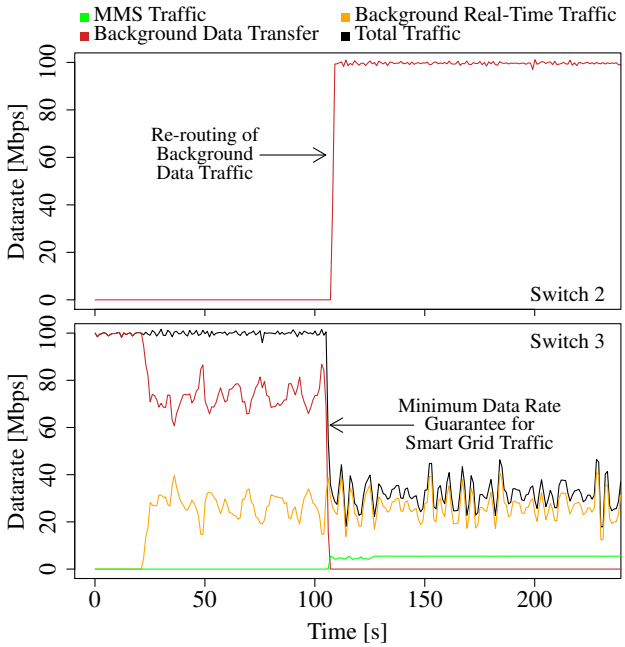

Fig. 5: Scenario 2a: Load Management at Switches 2 and 3

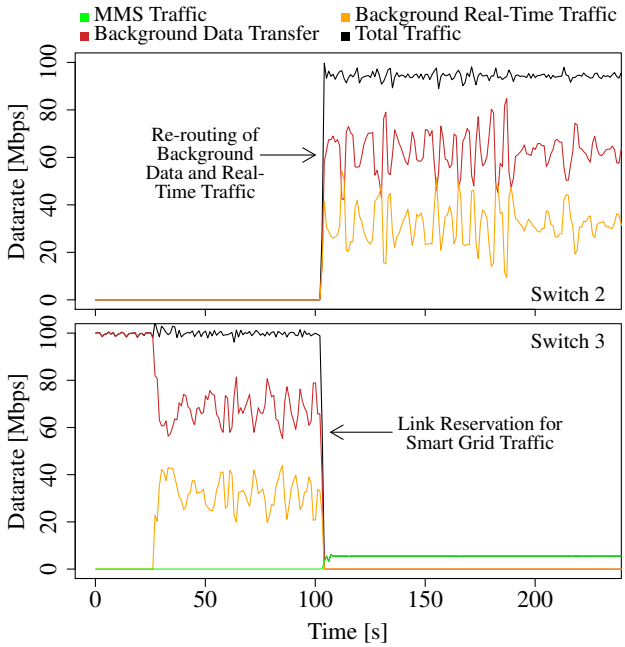

Fig. 6: Scenario 2b: Link Reservation at Switch 3

able to enqueue traffic flows using these queues, mapping the QoS requirements of different traffic classes being equivalent to priorities. The traffic class of an arriving packet is identified by matching against specified rules such as the packet's application layer protocol. Table I contains the traffic class along with assumptions on minimum data rate and latency requirements, considered for this experiment.
Both following QoS approaches have been evaluated by inserting successively the hereafter detailed traffic flows into the network (cf. Figure 2): background data transfer from Server 2 to Client 1 (TCP/FTP-based), background real-time traffic from Server 1 to Client 1 (UDP-based) and MMS reports from Server 2 to Client 1. In addition, the network is set to 100 Mbps maximum data rate, whereat 5 Mbps are reserved for network control at all times, leaving effective data rates of 95 Mbps.

Scenario 2a: Reserved Data Rate for Time-Critical Services

This QoS approach aims at reserving data rate for time-critical services by guaranteeing minimum data rates for each traffic flow and re-routing traffic flows with lower priority whenever the available data rate of a link might be exceeded. Therefore the SDN controller keeps track of all active traffic flows within the network along with their routes and priorities. For this approach, the following algorithm has been developed:

1) **Shortest Path Calculation:** First, the shortest route is calculated for arriving packets.
2) **Data Rate Demand:** For each link, the SDN Controller determines the QoS demand of the new flow and compares its associated minimum data rate with the available data rate of the link.
3) **Priority Grouping:** If the data rate is sufficient on all links the flow is added, otherwise the priority of the new flow and of those already active on the respective link is compared. Subsequently, flows are grouped into flows with higher or same priority and those with lower priority.
4) **Re-routing of New Flow:** If the link capacity is insufficient for the new and all higher/same priority flows, the second best route is calculated and tested, restarting at step (2).
5) **Sorting of Lower Priority Flows:** Else, lower priority flows are sorted into a list depending on their overlap with the route of the new flow and their priority, defining an order for re-routing.
6) **Determination of Lower Priority Flows for Re-Routing:** While link capacity is insufficient for higher, same and the remaining lower priority flows, the next lower priority flow is marked for re-routing and removed from the list.
7) **Unmodified Lower Priority Flows:** All flows remaining in the list maintain their current route.
8) **Calculation of Alternative Routes:** New routes are being calculated on a virtual topology, excluding the links of the new flow's route, for flows that have to be shifted. Afterwards, for each pair of flow and alternative route all steps starting from (2) have to be repeated.
9) **Drop of Lower Priority Flows:** If no alternate routes are available flows are dropped.
10) **Re-routing of Lower Priority Flows:** Else, OFFlowMod messages are prepared for establishing new flow entries at the switches for the lower priority traffic flows affected.

Figure 5 shows active transmissions and its corresponding data rates at Switch 2 (top) and 3 (bottom), when applying this approach. At the beginning, background data traffic is transmitted via Switch 3, exploiting all available data rate. After 30 s background real-time traffic is added to the network. Since the link between Switch 3 and 4 is used by this transmission as well, the data rate of the background data traffic is reduced accordingly. Next, background data traffic needs to be re-routed via Switch 1 and 2 in order to enable high priority MMS reports on the same link after 105 s. Otherwise, adding MMS reports would cause the sum of minimum data rate requirements to exceed the available capacity of the link between Switch 3 and 4, which is prevented by shifting the lowest priority traffic flow.

Scenario 2b: Dedicated Links for Time-Critical Services

For the second approach critical services have been specified, which are granted dedicated links for data exchange. Accordingly, the previous algorithm has been extended as follows:

1) **Route Reservation:** If the new traffic flow is a critical service, all links of its shortest route are reserved.
2) **Re-routing of Overlapping Flows:** All flows on these links are rerouted according to steps (8) to (10) (above).
3) **Blocking of Reserved Path:** The reserved route is blocked for other, new traffic flows, being excluded from subsequent shortest path calculations. Neither lower priority, nor other critical flows are allowed on reserved links.

Results gained from this approach are given in Figure 6, whereat Smart Grid monitoring has been defined as critical service. Again the data rate of background data traffic drops after background real-time traffic is added to the network. Yet, subsequent to the occurrence of MMS reports after 105 s, the controller re-routes both other traffic flows via Switch 1 and 2, reserving the shortest path for Smart Grid communication.

The results of Scenario 2a and 2b highlight seamless load shifting for QoS purposes on basis of SDN as old routes are not removed until the successful establishment of new paths. This approach might reach its full potential when combined with fast recovery, enabling traffic offloading to restore routes for the transmission of critical control commands of the power system and thus rendering additional dedicated backup links unnecessary. In addition, extensibility and compatibility can be identified as the most important strengtha of this SDN-enabled QoS mechanism compared to traditional approaches like MPLS.

V. Conclusion and Future Work

In this work, SDN has been applied for communications in Smart Grids, considering specific requirements of transmission and distribution level of the power system. Algorithms for enabling fast recovery and load management have been developed and evaluated using a SDN testbed with IEC 61850 compliant traffic. While mechanisms for rapid failover are of major importance for time-critical control commands in transmission power grids, both distribution and transmission power grid profit from QoS-based load management either with respect to public traffic or less prior grid services. It has been demonstrated that the reliability of communication networks for Smart Grids can be increased significantly compared to traditional mechanisms like OSPF and MPLS, as shown in Table II. In particular, the outstanding benefit of applying SDN to Smart Grid communications lies in the combination of the presented features and its openness towards introducing additional functionalities. Thus, power systems are offered new ways - like autonomous monitoring and advanced control - to deal with failures, transforming the overlayed ICT infrastructure into a self-healing system.

TABLE II: Comparison of Proposed SDN for Smart Grids Approach with Traditional Networking Solutions

	Fast Recovery	**QoS**	**Smart Grid Function Integrability**
Standard Routing	(X)		
MPLS	(X)	X	
SDN4SmartGrids	X	X	X

In future work, we will integrate algorithms for precise load balancing of the network in order to exploit the available capacity on all links in an optimal manner. One major challenge for the successful application of SDN is improving the reliability of the SDN controller, which currently poses a single point-of-failure. In order to overcome this, the controller could be run atop of a virtualized execution platform, applying appropriate fault-tolerance mechanisms such as periodic replication of the virtual machine's state or its migration in case of failure. Enhancing this idea, we aim at creating a holistic virtualization approach for the ICT infrastructure of substations. Thus, the SDN-enabled communication network will be combined with a virtualized execution platform for monitoring, protection and control applications, building upon previous analysis in [15]. Furthermore, economical consequences of the SDN approach in conjunction with Smart Grid communication are to be evaluated. Finally, future work will need to consider shared, heterogeneous communication infrastructures, the scalability of the SDN concept for extensive networks - i.e. control network latency - as well as implications and mechanisms for security - i.e. vulnerability of the control network via the switches.

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 SmartC2Net project, funded by the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. FP7-ICT-318023. Also, the authors would like to thank Michael Baumeister for his support.

References

[1] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart Grid - The New and Improved Power Grid: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 4, pp. 944–980, 2012.

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

[3] N. McKeown *et al.*, "OpenFlow: Enabling Innovation in Campus Networks," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

[4] Open Networking Foundation, *OpenFlow Switch Specification Version 1.0.0*, 2009. [Online]. Available: https://www.opennetworking.org/sdn-resources/onf-specifications/openflow.

[5] 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*, Toronto, Canada, 2013, pp. 558–563.

[6] A. Goodney, S. Kumar, A. Ravi, and Y. H. Cho, "Efficient PMU Networking with Software Defined Networks," in *IEEE International Conference on Smart Grid Communications*, Toronto, Canada, 2013, pp. 378–383.

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

[8] A. Sgambelluri, A. Giorgetti, F. Cugini, F. Paolucci, and P. Castoldi, "OpenFlow-Based Segment Protection in Ethernet Networks," *J. Opt. Commun. Netw.*, vol. 5, no. 9, pp. 1066–1075, Sep. 2013.

[9] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester, "OpenFlow: Meeting Carrier-grade Recovery Requirements," *Comput. Commun.*, vol. 36, no. 6, pp. 656–665, Mar. 2013.

[10] IEC TC57, *IEC 61850: Communication networks and systems for power utility automation*, Geneva, Switzerland: International Electrotechnical Commission.

[11] G. Andersson *et al.*, "Causes of the 2003 Major Grid Blackouts in North America and Europe, and Recommended Means to Improve System Dynamic Performance," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1922–1928, 2005.

[12] D. Erickson, "The Beacon OpenFlow Controller," in *HotSDN*, ACM, 2013.

[13] M. Zillgith, *libIEC61850 0.7.3*, 2013. [Online]. Available: http://www.libiec61850.com/libiec61850/.

[14] Y. Lei, C.-H. Lung, and A. Srinivasan, "A cost-effective protection and restoration mechanism for ethernet based networks: an experiment report," in *Workshop on High Performance Switching and Routing*, 2004, pp. 350–354.

[15] N. Dorsch, B. Jablkowski, H. Georg, O. Spinczyk, and C. Wietfeld, "Analysis of Communication Networks for Smart Substations Using a Virtualized Execution Platform," in *IEEE International Conference on Communications*, Sydney, Australia, Jun. 2014.