Reconfigurable Resource Allocation in Dynamic Transport Networks

Tobias Enderle
Universität Stuttgart, IKR
Stuttgart, Germany
tobias.enderle@ikr.uni-stuttgart.de

Norbert Ascheuer
atesio GmbH
Berlin, Germany
ascheuer@atesio.de

Uwe Bauknecht
Universität Stuttgart, IKR
Stuttgart, Germany
uwe.bauknecht@ikr.uni-stuttgart.de

Ulrich Gebhard
Nokia Bell Labs
Stuttgart, Germany
ulrich.gebhard@nokia-bell-labs.com

Karl Hübner
DCAITI, TU Berlin
Berlin, Germany
karl.huebner@dcaiti.com

Robert Protzmann
Fraunhofer FOKUS
Berlin, Germany
robert.protzmann@fokus.fraunhofer.de

Christian Raack
atesio GmbH
Berlin, Germany
raack@atesio.de

Arthur Witt
Universität Stuttgart, IKR
Stuttgart, Germany
arthur.witt@ikr.uni-stuttgart.de

***Abstract*—Future traffic patterns in transport networks will be much more volatile than today due to emerging applications like connected cars or the Internet of Things. The multitude and frequency of change of their connections requires a high level of automation for which an SDN-controlled network architecture is indispensable. We therefore present three SDN-based reconfiguration approaches covering QoS-aware routing, load balancing and optical band defragmentation. We show that the presented approaches are capable of handling such volatile traffic and are improving network efficiency.**

***Index Terms*—automotive data traffic, network reconfiguration, QoS, SDN, WDM**

I. Introduction

Network operators face significantly changing traffic patterns in their aggregation and transport networks not just due to the rapidly growing share of video streaming and faster access technologies like FTTx and 5G, but also due to novel participants and communication schemes such as the tactile Internet [1] and the Internet of Things (IoT). Especially services devised by the automotive sector are expected to have a large impact regarding both volatility and volume of traffic due to the diversity of car-based services and the inherent mobility of their end points. Some of these services have strict quality of service (QoS) requirements regarding latency which necessitates data centers closer to the network edge. The multiplicity and dynamicity of these services call for a high degree of network automation, therefore requiring a software-defined networking (SDN) architecture that is capable of quickly allocating a multitude of services across network layers end-to-end and under QoS constraints.

An SDN controller for a dynamic IP-over-WDM network needs to find a network configuration in terms of optical circuits and traffic routing with respect to the current resource state and the QoS constraints of new and existing services. This allows for an optimal resource utilization, but the associated circuit setup times preclude an instantaneous reconfiguration. For more immediate changes, highly reactive adaptions on short time scales can be achieved by shifting traffic between paths on the IP-layer which allows to maintain network stability by performing a globally optimized load balancing. Finally, any reconfiguration action on the optical layer needs to be broken down into a sequence of serialized actions representing a migration path in order to avoid conflicts and minimize potential disruptions to the network.

This work has been performed in the framework of the CELTIC EUREKA project SENDATE-TANDEM (Project ID C2015/3-2), and it is partly funded by the German BMBF. The authors alone are responsible for the content of the paper.

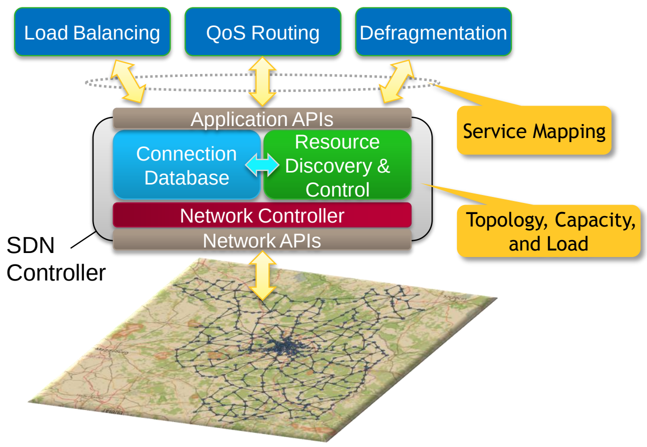

Fig. 1. SDN architecture including Berlin-Brandenburg topology.

We propose several algorithmic approaches to efficiently solve these specific sub-problems: (i) a QoS routing engine ensures that service requests with critical requirements (e.g., guaranteed bandwidth, low latency) are routed on a path that just fulfills the QoS requirements while leaving the network with the highest flexibility to satisfy subsequent service requests, (ii) a highly reactive load-balancing module rearranges IP routes with the goal to maximize throughput and to avoid congestion. The module also guarantees survivability of the (reconfigured) service routings at the fiber layer and (iii) a defragmentation module provides a sensitive migration path to rearrange optical circuits to free wavelength resources. We evaluate the impact of reconfigurations in prototypical network scenarios including highly volatile automotive traffic and edge data centers. The overall architecture is summarized in Fig. 1.

II. Network and Dynamic Traffic Scenario

The reconfiguration studies presented in the following are based on a common network and dynamic traffic model developed in [2]. A wavelength-division multiplexing (WDM) topology in the Berlin-Brandenburg area has been precomputed using presumed central offices as points of presence (PoPs) and embedding WDM half-rings into a realistic fiber-cable layer, see [2, Fig. 2]. It features 728 WDM links and 534 PoPs and is assumed to be flexible enough to realize any desired IP topology by terminating or forwarding any wavelength channel carried over the connected fibers using reconfigurable optical add-drop multiplexers (ROADMs).

The traffic model from [2] establishes service classes from fixed network access as well as from connected cars and data center synchronization. Traffic from automotive applications turns out to introduce a dramatic increase in volatility (in time and space), see [2, Fig. 4], and in QoS requirements such as latency and jitter sensitivity depending on the service class, see [2, Table I]. Also, certain applications will require strict QoS guarantees, leading to a shift of workload from centralized data centers to smaller data centers at the network edge.

For this study we combined automotive and fixed access service classes from [2] based on their delay requirement into 4 groups, namely *auto-low* and *fixed-low* with a delay requirement of 10 ms and *auto-medium* and *fixed-medium* with a delay requirement of 1000 ms. The assigned delay requirements are end-to-end propagation delays in the transport network. Delays that arise between the vehicle and the edge of the transport network at the air interface were estimated with the help of [3] and subtracted from the delay requirements given in [2].

Fig. 2 shows the resulting total traffic amount during a day varying between 10 and 24 Tbps. The share of low-delay traffic varies between 20 and 39 %.

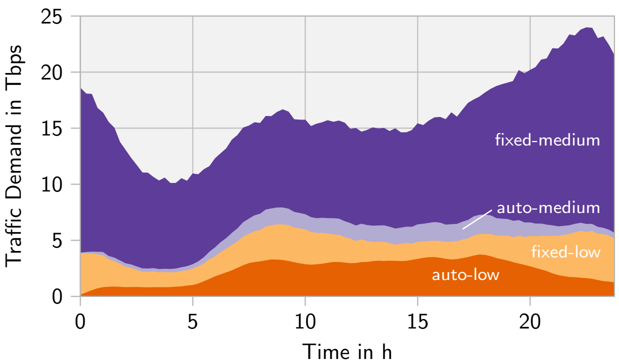

Fig. 2. Traffic amount and composition for the Berlin-Brandenburg network during 24 hours (stacked).

III. QoS Routing Engine

Routing along shortest paths is favorable in many situations. However, considering for example propagation delay, routing services without strict delay requirements on shortest paths can congest links in the network while longer paths over less utilized links could be used as well. Additionally, short paths often overfulfill QoS requirements like delay. It is therefore desirable to route services that have less strict delay requirements on longer paths for two reasons. First, to avoid overutilization of highly loaded links and second, to deliver service delays with little overfulfillment to provide an incentive for the customer to buy a more expensive service class if he needs lower delay.

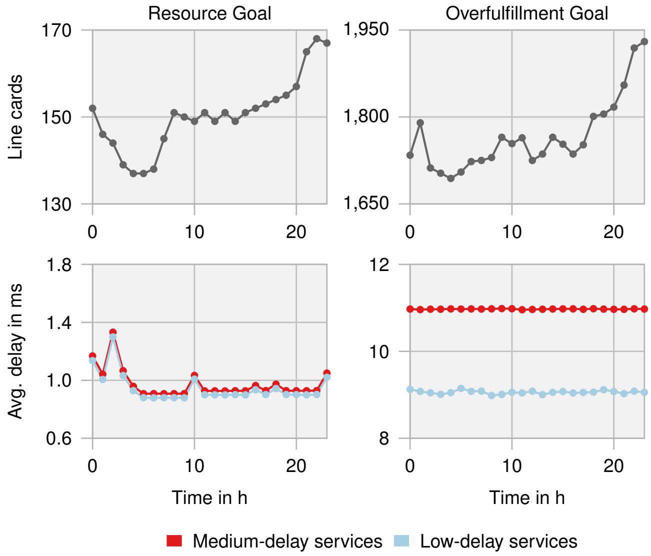

Fig. 3. Active line cards (top) and average propagation delay of services (bottom) for two corner points of the QoS routing approach.

We propose a reconfiguration approach that optimizes the traffic routing in the network across packet and optical layers. The optimization goal is an adjustable trade-off between delay overfulfillment and required hardware, e.g., router line cards. In the face of depleting resources, the algorithm avoids blocking as long as possible. We solve this optimization problem with the help of a metaheuristic-based framework.

In Fig. 3 we show the amount of active line cards and the average propagation delay of the services for two corner points of our approach using the scenario described in Section II. In the optimization with the *Resource Goal* the amount of line cards is minimized providing maximum flexibility for the routing of subsequent service requests. For the *Overfulfillment Goal* the delay overfulfillment of the low-delay services is minimized, i.e., the delay is close to 10 ms, while the medium-delay services are routed with a delay that is strictly higher than 10 ms. Due to the compact geographical layout long paths are necessary to achieve these high delays resulting in a large amount of active line cards. However, the trade-off between these two corner points is adjustable. Furthermore, the reduction of overfulfillment can be applied to only a subset of the demands. This allows operators to flexibly adapt the approach to their requirements and business models.

IV. Reconfiguration to Balance IP Load

The main target of this use case is to determine reconfiguration steps in order to avoid highly utilized (IP) links ([4], [5]). These links might result in bottlenecks in the system and cause traffic loss. Whenever the load on a single IP link exceeds a certain configurable threshold (e.g., 70 %) the SDN controller triggers an optimization engine that tries to reroute services in order to minimize the maximum utilization of an IP link.

The optimization engine consists of a path-selection based mixed-integer programming model, which selects from a given set of potential routing paths those that (i) minimize the maximum utilization of the IP links, (ii) keep the protection paradigm, i.e., in the case of 1+1-routing the selected IP paths have to be disjoint on the WDM layer, (iii) shorten

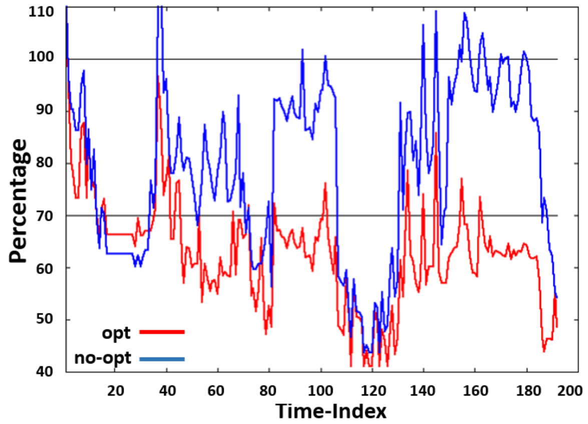

Fig. 4. Maximum relative load of an IP link over one day (blue line: no reconfiguration, red line: reconfiguration triggered when IP utilization of at least one link exceeds 70 %).

routing-paths if it is considered to be useful, (iv) minimize (or bound) the total number of path changes in order to reduce the migration effort. The path-engine that is initialized beforehand and updated whenever needed (e.g., in the case of link-failures) is configured such that only paths are considered that fulfill all technical side constraints, such as latency, delay, etc.

Using the scenario setting described in [2] the optimization engine was compared to a non-reconfiguration-scenario on a time interval of 24 hours (see Fig. 4). Note, that in the non-reconfiguration case the maximum load exceeds the 100 % load a couple of times (i.e., traffic will be lost). Moreover, reconfiguration significantly reduces the maximum link load. Optimization results are obtained in short computing times that would be suitable for use in real-world SDN controllers.

V. Reconfiguration for Optical Band Defragmentation

Defragmentation involves two steps: (i) finding better routes for current wavelength services, and (ii) migrating the current routing to the new one. Due to dependencies, some paths must be torn down before others can be set up. In the simple migration example from Fig. 5(a) to Fig. 5(b) wavelength services S1 and S2 must be rerouted before wavelength services S3 and S4 can be moved to their target path.

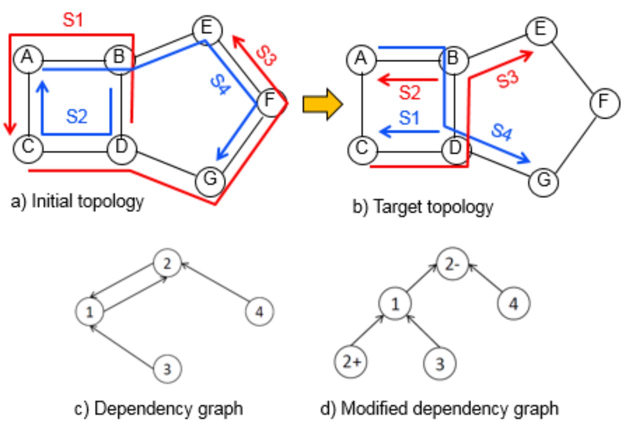

Fig. 5. Migration scenario and dependencies.

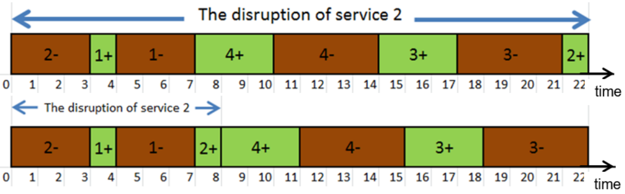

Fig. 6. Disruption time of service 2 for two different sequences.

This is formulated in the dependency graph in Fig. 5(c) where services are represented as vertices and dependencies as edges [6]. The loop in Fig. 5(c) can be resolved by splitting nodes with bidirectional dependencies (here: node 2) in a vertex v^- for incoming and a vertex v^+ for outgoing dependencies. The resulting modified dependency graph is shown in Fig. 5(d). For this graph two possible migration sequences are shown in Fig. 6, assuming that the time for withdrawal and establishment of a service is equal to its path length in hops. In large networks an exhaustive analysis would be too complex due to the high number of possible sequences. Instead, we developed a genetic algorithm [7] to find the sequence with minimum overall service disruption time. For two smaller networks, we verified that our algorithm can find good solutions by comparing them with an exhaustive analysis.

VI. Conclusions

In this paper we presented an SDN tool set and as a showcase presented three application examples. This tool set is based on a realistic network scenario exceeding the scale of other studies in terms of complexity and size.

The presented QoS routing approach reduces the delay overfulfillment and the amount of required resources in scenarios with distinguished service classes. We have seen that it was possible to significantly reduce link overload applying an optimization-based load balancer and a defragmentation algorithm can significantly reduce service disruption time. Both approaches are capable of handling even complex network scenarios in a very short time frame.

Acknowledgment

We appreciate the fruitful discussions with Frank Ilchmann (formerly Nokia Bell Labs, Stuttgart), his suggestions and the work he contributed to the project.

References

[1] M. Simsek, A. Aijaz, M. Dohler, J. Sachs, G. Fettweis, "5G-Enabled Tactile Internet," IEEE Journal on Selected Areas in Communications, vol. 34, no. 3, pp. 460–473, March 2016.
[2] R. Protzmann, K. Hübner, N. Ascheuer, U. Bauknecht, T. Enderle, U. Gebhard, C. Raack, A. Witt, "Large-scale Modeling of Future Automotive Data Traffic towards the Edge Cloud," accepted for publication at Photonic Networks; 20th ITG-Symposium, May 2019.
[3] 3rd Generation Partnership Project, "3GPP specification: 23.203; Group Services and System Aspects; Policy and charging control architecture (Release 15)," Version 15.4.0, September 2018.
[4] A. Bley, "An Integer Programming Algorithm for Routing Optimization in IP Networks," Algorithmica, 60(1):21–45, 2011.
[5] R. Klähne, "Models for Survivable Multi-layer Network Reconfiguration," Master's thesis, Technische Universität, Berlin, Germany, 2009.
[6] G. Smith and R. Walford, "The Identification of a Minimal Feedback Vertex Set of a Directed Graph," IEEE Transactions on Circuits and Systems, vol. 22, no. 1, pp. 9–15, January 1975.
[7] C. Moon, J. Kim, G. Choi, Y. Seo, "An efficient genetic algorithm for the traveling salesman problem with precedence constraints," European Journal of Operational Research, vol. 140, no. 3, pp. 606–617, 2002.