Electronic Communications of the EASST
Volume 080 (2021)

Conference on Networked Systems 2021 (NetSys 2021)

Towards Deterministic Reconfigurable Networks

Zikai George Zhou

5 pages

Guest Editors: Andreas Blenk, Mathias Fischer, Stefan Fischer, Horst Hellbrueck, Oliver Hohlfeld, Andreas Kassler, Koojana Kuladinithi, Winfried Lamersdorf, Olaf Landsiedel, Andreas Timm-Giel, Alexey Vinel
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Towards Deterministic Reconfigurable Networks

Zikai George Zhou[1] *

[1] george.zhou@tum.de,
Chair of Communication Networks
Technical University of Munich

Abstract: Compared with legacy networks, programmable networks are highly flexible and need to be reconfigured dynamically. In this early work paper, we study the fast and consistent network update which is the key enabler to realize deterministic reconfigurable networks. The reconfiguration speed is one side of the coin. The ongoing best-effort traffic cannot be interrupted during the network reconfiguration as well. In terms of reconfiguration speed, we implement and compare our method with the state-of-the-art decentralized and centralized update methods.

Keywords: Reconfigurable Networks, Network Measurements, Programmable Data Plane with P4

1 Introduction

Communication networks have become a critical infrastructure and continuously increase in scale and complexity. In order to adapt the massive amount of devices and services, and to tune according to policy, networks are required to be re-configured or updated dynamically. The approach of centrally controlling networks can prevent many oscillations brought by distributed route computation [JLG+14]. Meanwhile, since the update schedule is computed at the centralized controller, it also brought multiple challenges when updating the network.

The first challenge is how to roll out updates efficiently and correctly. The efficiency is to realize shortest update time. The correctness is to maintain consistency properties e.g., blackhole-freedom, loop-freedom, and congestion-freedom during the update [FSV19]. However, operators commonly have to make a trade-off between these two goals. Even worse, if the controller has an inconsistent view to the network or inaccurate state of the data plane, the consistency properties during update could be violated e.g., forwarding loop or link congestion.

The second challenge is to ensure that deterministic transmission should not be violated during the update, especially for those time-triggered networks such as Industrial Control Networks and Industrial Internet of Things (IIoT). The deterministic transmission reflects in predictable end-to-end latency [VĐZ+19] and bounded number of frame loss [LWP+19].

To make the update time short, we propose and implement a P4Update framework which make the data plane update fast and provably consistent. A local verification scheme is proposed to solve the first challenge with the help of programmable data plane with P4 [BDG+14]. The data plane state such as graph information maintained by the controller is pushed to and temporarily stored by programmable switches. Probe packets carrying the real-time data plane

*

state are used to compare with the controller's view. The data plane update is triggered by the verification result. To demonstrate the update efficiency, we emulate our approach and compare with a decentralized update scheme ez-Segway which showcases the bottleneck of update speed [NCC17]. We prove that our approach maintains the consistency without the interactions with the centralized controller during the update.

Before solving the second challenge, we study and measure the predictability of static network performance. In particular, we build a testbed setup with NIC in the end stations and Time Sensitive Networking (TSN) evaluation switches as the intermediate nodes. Both the application to application layer delay and the NIC to NIC delay are measured.

2 Motivation

Software-Defined Network (SDN) has brought tremendous benefits, such as high performance in throughput, high utilization, and high data plane availability [FGH+21]. Due to the limitations of adopting new protocols, Programmable Data Plane (PDP) has been born and evolved continuously [HHM+21]. On one hand, programmable networks has shown great flexibility through centrally managing the networks [HAB+19]. On the other hand, some common challenges occur during reconfiguring the network, such as consistency properties violation and un-deterministic real-time Quality of Service (QoS), Service Level Agreement (SLA) violation [FSV19]. Solving these challenges became necessary especially when many networks exist in a constant state of change. Operators of ISP must update the data plane state frequently independent of their optimization goal [JLG+14]. The dynamic behavior of IIoT requires the time-triggered schedules in TSN to adapt to dynamic updates while maintaining the strict latency requirements [LWP+19].

To meet these requirements and solve these challenges in reconfigurable networks, we must firstly know what is going on in the current networks. Often network measurements are the fundamental tasks of network control and management. The operators need to measure network performance continuously to ensure QoS to their customers [YRS15]. Moreover, they also need to troubleshoot their network in case there are some failures or misconfigurations [HHJ+14]. Along the timeline, the network measurements could be classified into traditional network measurements, software-defined network measurements and emerging network telemetry. Traditional network measurements have been widely used in the network management due to ease of deployment, but it lacks of accuracy and affects the network state [Mor16]. With the emergence

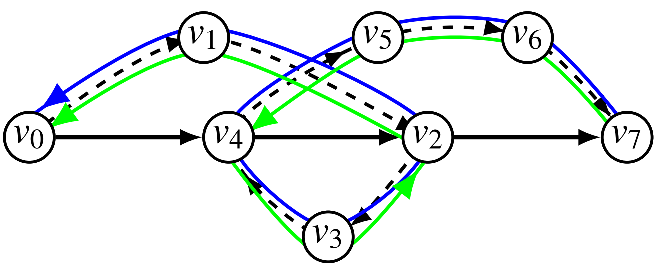


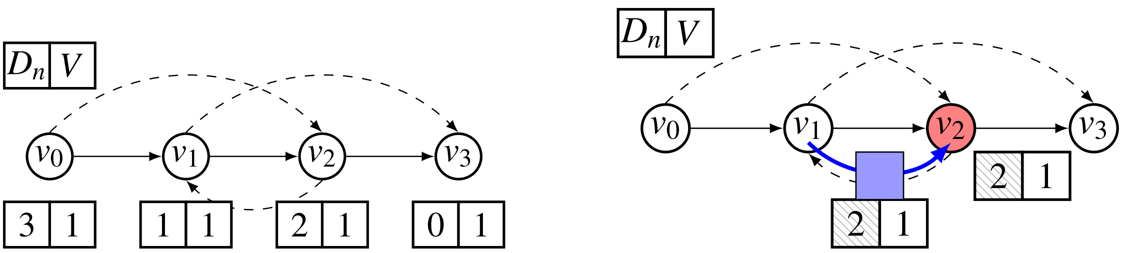
Figure 1: Illustration of SL- and DL-P4Update. The blue line represents the update in SL-P4Update and the green lines show the DL-P4Update segmentation.

of SDN and PDP, some software-defined measurement schemes have been proposed which improves measurement openness and transparency [YLS+15]. Network Telemetry was becoming a popular technology to make fine-grained measurements to the current network state [Yu19].

3 Towards Consistent Network Updates in Programmable Networks with P4

When implementing network updates, operators commonly choose a trade-off between update speed and consistency. In order to ensure the consistency properties, probe packets are generated at the data plane to provide two functions.

The first one is to coordinate between data plane. The Figure 1 is used to demonstrate the coordination process. The network flows are segmented according to the old (solid line) and new routing paths (dashed line). In most update scenarios in which the old and new paths are disjoint, we adopt the Single-Layer (SL) approach. $V7$ is the egress node, using the SL approach, the probe packets are sent back from $V7$ to $V0$ along the blue line. The main idea is to avoid complicated coordination procedures. In order to improve the update speed, we design a dual-layer update mechanism. The Dual-Layer (DL) approach is applied to complicated scenario which could implement parallel updates of multiple segments. Using the DL approach, $V0$ to $V2$ and $V4$ to $V7$ are two forwarding segments which could be updated in parallel.

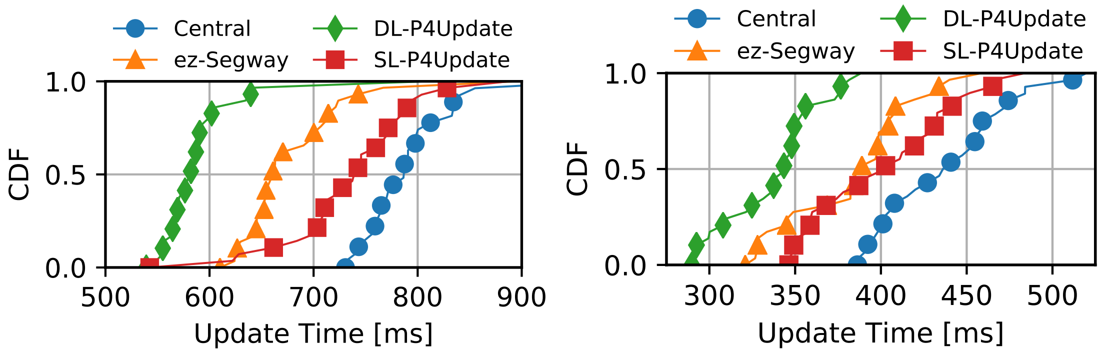


(a) No inconsistencies. (b) Erroneous distance.

Figure 2: Verifying update consistency, where the probe packets are forwarded via v_3, v_1, v_2, v_0 to update from the solid old path to the dashed new path. Scenario (a) shows a successful update without inconsistency issues, whereas (b) shows one example with inconsistent update information.

The second one is to verify the consistency between centralized controller and data plane state. The Figure 2 is used to demonstrate the verification process. The routing configuration version and distance of each node to the destination are stored in the probe packets. For example, the version of solid and dashed path is 1 and 2 separately. The distance of $V3, V1, V2, V0$ is $0, 1, 2, 3$ separately. From controller's perspective, if $V2$ contains the same distance as $V1$ in Figure 2b, blackhole or loop could occur.

To demonstrate the update speed, we implement two approaches of P4Update and ez-Segway using P4 and evaluate using BMv2 switch target [Con16] in typologies of B4 and Internet2. The centralized update method is used as the baseline. We measure the total update time of a single



(a) B4.

(b) Internet2.

Figure 3: Total update time CDF

flow which is from the timestamp the controller sending the new configuration to the timestamp the controller receiving the last update finish notification. The total update time of DL-P4Update is 15% and 14% less than the ez-Segway. The reason is that DL-P4Update involves fewer rounds of communication with controller and involve higher degrees of parallelism.

4 Future Work and Outlook

The reconfiguration mechanisms should be considered to adapt into the programmable hardware. The main challenge is about the hardware limitations, such as limited access times of stateful objects. Moreover, more measurement studies during the reconfiguration are considered to be done. But gathering and understanding the real-time data plane state is not trivial. Moreover, today's network measurement mechanisms are mostly done for offline analysis. Using online network measurement techniques to help ensure the stable performance during the network reconfiguration is also unknown. As a long-term goal, a closed-loop measurement, verification and control knob for reconfigurable networks is treated as the working direction.

References

[BDG+14] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese et al. P4: Programming protocol-independent packet processors. *ACM SIGCOMM Computer Communication Review* 44(3):87–95, 2014.

[Con16] T. P. L. Consortium. P4 behavioral-model. https://github.com/p4lang/behavioral-model/, 2016.

[FGH+21] A. D. Ferguson, S. Gribble, C.-Y. Hong, C. E. Killian, W. Mohsin, H. Muehe, J. Ong, L. Poutievski, A. Singh, L. Vicisano et al. Orion: Google's Software-Defined Networking Control Plane. In *NSDI*. Pp. 83–98. 2021.

[FSV19] K. Foerster, S. Schmid, S. Vissicchio. Survey of consistent Software-Defined Network updates. *IEEE Communications Surveys & Tutorials* 21(2):1435–1461, 2019.

[HAB+19] M. He, A. M. Alba, A. Basta, A. Blenk, W. Kellerer. Flexibility in softwarized networks: Classifications and research challenges. *IEEE Communications Surveys & Tutorials*, 2019.

[HHJ+14] N. Handigol, B. Heller, V. Jeyakumar, D. Mazières, N. McKeown. I know what your packet did last hop: Using packet histories to troubleshoot networks. In *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*. Pp. 71–85. 2014.

[HHM+21] F. Hauser, M. Häberle, D. Merling, S. Lindner, V. Gurevich, F. Zeiger, R. Frank, M. Menth. A Survey on Data Plane Programming with P4: Fundamentals, Advances, and Applied Research. *arXiv preprint arXiv:2101.10632*, 2021.

[JLG+14] X. Jin, H. H. Liu, R. Gandhi, S. Kandula, R. Mahajan, M. Zhang, J. Rexford, R. Wattenhofer. Dynamic scheduling of network updates. In *SIGCOMM*. Pp. 539–550. ACM, 2014.

[LWP+19] Z. Li, H. Wan, Z. Pang, Q. Chen, Y. Deng, X. Zhao, Y. Gao, X. Song, M. Gu. An enhanced reconfiguration for deterministic transmission in time-triggered networks. *IEEE/ACM Transactions on Networking* 27(3):1124–1137, 2019.

[Mor16] A. Morton. Active and Passive Metrics and Methods (with Hybrid Types In-Between). RFC 7799, May 2016.
doi:10.17487/RFC7799
https://rfc-editor.org/rfc/rfc7799.txt

[NCC17] T. D. Nguyen, M. Chiesa, M. Canini. Decentralized consistent updates in SDN. In *Proceedings of the 2017 ACM Symposium on SDN Research (SOSR)*. Pp. 21–33. 2017.

[VĐZ+19] A. Van Bemten, N. Ðerić, J. Zerwas, A. Blenk, S. Schmid, W. Kellerer. Loko: Predictable latency in small networks. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*. Pp. 355–369. 2019.

[YLS+15] C. Yu, C. Lumezanu, A. Sharma, Q. Xu, G. Jiang, H. V. Madhyastha. Software-defined latency monitoring in data center networks. In *International Conference on Passive and Active Network Measurement*. Pp. 360–372. 2015.

[YRS15] A. Yassine, H. Rahimi, S. Shirmohammadi. Software defined network traffic measurement: Current trends and challenges. *IEEE Instrumentation & Measurement Magazine* 18(2):42–50, 2015.

[Yu19] M. Yu. Network telemetry: towards a top-down approach. *ACM SIGCOMM Computer Communication Review* 49(1):11–17, 2019.