



Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

Robust hierarchical control plane for Transport Software-Defined Networks[☆]



Rafael B.R. Lourenço^{a,*}, S. Sedef Savas^a, Massimo Tornatore^{a,b}, Biswanath Mukherjee^a

^a University of California, Davis, USA

^b Politecnico di Milano, Italy

ARTICLE INFO

Keywords:

Hierarchical control plane
Resilient switch assignment and controller/orchestrator placement
Robust Transport Software-Defined Networks
Disaster survivability
Post-disaster restoration

ABSTRACT

Software-Defined Networking (SDN) enables the separation of data and control planes. Today, it is in use in several practical networks. Research on SDN is yet to focus on Transport Networks in a significant way. To provide efficient services at low cost for future technologies, such as 5G and beyond, the Transport Network will need to support network functions; adapt the network to different applications' traffic; provide high bandwidth at low latencies; integrate with other networks; etc. By fulfilling these requirements, a Transport Software-Defined Network (T-SDN) can become a fundamental part of the telecommunication infrastructure. However, most current SDN solutions have been developed for Layer 3, and cannot be directly applied to a T-SDN without proper adaptations, as Transport Networks have several characteristics that are different from those of Layer 3 networks. In particular, the design of a T-SDN control plane must address heterogeneity in terms of protocols and administrative network areas; as well as high reliability. A hierarchical control plane is suitable to support these characteristics. Accordingly, in this study, we analyze how to design a robust hierarchical control plane for T-SDNs. We discuss how resiliency against random failures can be provided through redundancy; and how survivability against correlated failures (such as disasters) can be achieved by effectively choosing network nodes where to place control-plane elements, and deciding how to route control-plane traffic. We formulate an Integer Linear Program to design a hierarchical, failure- and disaster-resilient T-SDN control plane. We also propose a heuristic for post-failure switch-controller reassignment. We compare our model with a disaster-unaware control-plane design whose objective is to reduce network-resource utilization. Our results show that we can achieve much higher disaster and failure resiliency, at the cost of slightly larger network-resource utilization.

1. Introduction

Software-Defined Networking (SDN) has gained a lot of popularity in recent years as several practical implementations demonstrated performance improvements and lower costs when compared to traditional networks [2,3]. Throughout different protocols and implementations, the common characteristic of SDN is the decoupling of control and data planes.

The data plane of SDNs (in charge of data transmission) is comprised of switches; while the control plane is composed of controllers and other control elements. This separation gives the control plane a network-wide perspective that makes it easier for resources to be managed in creative ways [4,5]. This enables operators to provide better services

(latency, bandwidth, etc.) at lower costs.

So far, most SDN-related systems, standards, and solution proposals have focused on Layer 3 of the communication stack, which does not include much of the Transport Network infrastructure. However, two main factors now motivate the use of SDN in Transport Networks: *i*) operators' thinning profit margins; *ii*) evermore bandwidth-hungry, reliability-dependent, and delay-sensitive applications, such as Virtual Reality, Internet of Things, Ultra-High-Definition video formats, etc. These elements also impact other networks. For example, one tool that the future 5G standard (and beyond) will use to deal with these factors is cooperation between different networks. Such cooperation could also be facilitated by Transport Software-Defined Networks (T-SDN) —

[☆] A very preliminary version of a part of this work was presented at the High-Performance Switching and Routing (HPSR) 2015 conference in Budapest [1].

* Corresponding author.

E-mail addresses: rlourenco@ucdavis.edu (R.B.R. Lourenço), ssavas@ucdavis.edu (S.S. Savas), tornator@polimi.it (M. Tornatore), bmukherjee@ucdavis.edu (B. Mukherjee).

<https://doi.org/10.1016/j.osn.2018.05.004>

Received 31 January 2018; Received in revised form 16 April 2018; Accepted 19 May 2018

Available online 22 May 2018

1573-4277/© 2018 Elsevier B.V. All rights reserved.

which is another motivation for the use of SDN in Transport Networks.

However, some characteristics specific to Transport Networks hamper a direct use of much of the already-acquired SDN knowledge without proper adjustments. One of the most distinctive characteristic is the heterogeneity of coexisting protocols, architectures, and systems (sometimes specific to each layer) of a Transport Network. In addition, several network element manufacturers differentiate themselves from their competition by introducing proprietary technologies and management systems, which are usually incompatible with other manufacturer's solutions. All this heterogeneity is common in most current operational Transport Networks, and it is evidenced by each network being composed of several administratively-independent *network areas* [6]. This reality contrasts with Layer-3-based SDN, where elements are usually much more homogeneous and inter-operable.

To deal with such heterogeneity, T-SDN efforts have been focusing on a solution where the control plane is structured hierarchically [7–9]. The lower level of this structure is composed of controllers, each capable of managing elements in a certain network area. Controllers, then, communicate with a centralized entity, named orchestrator, so that the network may provide services across different areas.¹ Such controllers and orchestrators can either be physical machines or virtualized systems [10].

Another particular characteristic of Transport Networks is that they are built to be extremely reliable. Many long-distance operators offer services at five 9s availability (sometimes higher [11]). Such highly-reliable systems must survive not only single and double failures (and more) but also large, correlated failures, such as disasters.

Achieving very high reliability in SDN is very challenging. The decoupling of data and control planes introduced by SDN represents a break in the long-standing fate-sharing principle of control and data planes of traditional network elements: i.e., it is possible for SDN data-forwarding elements to lose contact with control elements, and vice versa. This loss of control might lead network elements to be unpredictable and in unstable states. Thus, a major contingency for the successful utilization of SDN concepts in the availability-demanding Transport Network is the resiliency and the survivability of the SDN control plane and its communication with data-plane elements.

In this study, we propose a method to design a robust, hierarchical control plane for T-SDNs. To deal with heterogeneity, our solution assigns switches of different network areas to respective controllers, decides how many and where to place controllers in each area, and connects them to a central, primary orchestrator (and also backup orchestrators), in a hierarchical architecture. Our method also decides how to route switch-to-controller and controller-to-orchestrator traffic, avoiding shortest paths when more reliable, longer paths are also within latency bounds. To provide high reliability, our solution: *i*) provides redundancy of control-plane elements for failure resiliency (i.e., multiple controllers, backup orchestrators, and paths connecting them); *ii*) efficiently places controllers and orchestrators in the network (while choosing what paths to connect them), to minimize the effects of disasters. Our method provides all of the above while respecting other system-related requirements (such as maximum control traffic latencies, controller capacities, etc.).

The method we propose is mathematically formulated into an Integer Linear Program (ILP) that takes less than a few minutes to execute for a US-wide, 24-node network (running on an Intel Core i7, 16 GB RAM machine). Our model is capable of designing a robust control plane; however, severe, unpredicted failures might still disrupt it. Because of that, we also propose a simple heuristic to instruct how to proceed in case such a disrupting failure occurs. Our results show that our *disaster-aware* control plane is much more reliable than

¹ Other reasons for a hierarchical control plane include: backwards compatibility with brownfield Transport Networks; higher resiliency due to the distributed control plane; scalability; etc.

a *disaster-unaware* control plane designed to simply minimize control traffic latency, at the cost of slightly higher resource utilization.

The rest of this work is organized as follows: Section 2 briefly reviews the literature related to SDN control-plane design. Section 3 investigates the characteristic of the T-SDN hierarchical control plane. Section 4 details how we achieve high reliability in T-SDN control plane. Section 5 formally describes our problem, and provides a mathematical formulation and a heuristic in case of failures. Section 6 discusses the lessons learned using practical numerical examples. Section 7 concludes the study.

2. Related work

In this section, we briefly review contributions to the general problem of placing controllers in an SDN.² Then, we discuss studies that focus on the relationship between where controllers are placed and network resiliency.³

2.1. Controller placement

The Controller Placement Problem (CPP) was first proposed in Ref. [13]. Several metrics (such as worst and average case switch-to-controller latency, and number of switches reachable from each controller) and how they affect the optimal location of controllers in a network were considered in this study. The CPP was further investigated in Ref. [14], when they introduced constraints on the controllers' capacity to manage network elements, calling it the Capacitated CPP (or CCPP).

The placement of controllers in Wide-Area Networks (WAN) was studied in Ref. [15]. The authors proposed to first partition the WAN into different components through a clustering algorithm; and, second, decide where to place controllers for each component. This approach is not suitable for brownfield Transport Networks, as the different areas in these environments have usually evolved organically through the years, and it is very costly to completely re-define them.

In a mobile cloud-computing environment, a two-tier control plane is presented in Ref. [16]. The authors propose a solution where a centralized SDN orchestrator monitors and decides when and where to place local controllers depending on the network state. Other works that study some variation of the CPP include: Ref. [17] which investigated a minimum-cost solution to the CPP while limiting the maximum switch-to-controller latency; methods to dynamically grow/shrink the number of controllers according to the network state were proposed in Refs. [18,19]; practical implementations of distributed control planes were presented in Refs. [20–22].

Although distributing the control plane is a first step towards resiliency in SDNs, as it avoids a single point of failure, the works above have not specifically aimed at providing highly-reliable solutions — the focus of our study.

2.2. Controller placement and network resiliency

The Fault Tolerant Controller Placement Problem was introduced in Ref. [23]. The authors proposed a solution to achieve five-nines reliability of the switch-to-controller communication, with link, switch, and controller failures.

In Ref. [24], the authors proposed an approach that finds controller locations such that multiple possible switch-to-controller paths

² In our study, we refer to the dual problem of deciding where to place controllers and what switches to assign to them as a single problem, because these two aspects are intrinsically connected.

³ We understand *reliability* as the metric which expresses the percentage of the control plane (and its interactions within itself and with switches) that is not lost due to failures. *Resiliency* is the capacity of the control plane to sustain failures while remaining operational (partially or fully) [12].

Download English Version:

<https://daneshyari.com/en/article/6888430>

Download Persian Version:

<https://daneshyari.com/article/6888430>

[Daneshyari.com](https://daneshyari.com)