Abstract—Recent control plane solutions in a software-defined network (SDN) setting assume physically distributed but logically centralized control instances: a distributed control plane (DCP). As networks become more heterogeneous with increasing amount and diversity of network resources, DCP deployment strategies must be both fast and flexible to cope with varying network conditions whilst fulfilling constraints. However, current approaches are slow and only focus on controller placement, sometimes in combination with bandwidth or delay constraints. We demonstrate the capabilities of our optimization framework [1]–[3] for fast deployment of DCPs, emphasizing control service reliability, bandwidth and latency requirements. We show that the approach can produce robust deployment plans under changing network conditions. Compared to state of the art solvers, our approach is magnitudes faster, enabling fast DCP deployment within minutes and seconds rather than days and hours.

Index Terms—distributed control planes, elasticity, fault tolerance, reliability, software defined networks

I. INTRODUCTION

Distributing the control plane (CP) instances in a SDN setting can enhance the CP scalability and reliability: it decreases control latency, prevents overload of single controllers and assures robustness against controller failures. However, it comes at the cost of in-band control-traffic volume for coordinating distributed instances to appear as one logical entity. Finding a feasible DCP solution is a difficult task, mainly due to two major challenges: i) placing the control instances in such a way as to satisfy reliability and scalability constraints (controller placement), and ii) verifying that the introduced control-traffic in the network can be scheduled and routed with respect to bandwidth and latency requirements (routability). Controller placement involves estimation of the number and location of control instances. In general, it is NP-hard. Most DCP research prior to [1], [2] has mainly focused on i). However, assuring routability is key for guaranteeing optimal DCP deployment, especially since control-traffic must share resources with other network traffic, such as time-critical services. Furthermore, current approaches require long time (e.g., days) to find a feasible solution.

We presented a novel optimization approach in [1]–[3], which solves both i) and ii) extremely fast. The running time to guarantee routability in line with set performance requirements can for mid-sized and large topologies be reduced to minutes (and seconds) while still producing close to optimal solutions [1], in contrast to hours and days when using direct solvers such as CPLEX. In practice, this is a fundamental step towards enabling DCP elasticity, which is crucial for effective utilization of networked infrastructure resources and service connectivity.

II. OPTIMIZATION FRAMEWORK

The overall approach is a generic black-box optimization framework consisting of four main steps [1], [2]. We assume a programmable network consisting of commodity hardware, where networking nodes can be a controller and/or aggregator (a.k.a. switch). First, the mapping step determines the required number and location of controller instances (Fig. 1b). The association step associates remaining nodes (aggregators) to controllers (Fig. 1c). Third, the traffic-estimation step estimates the flow demands based on the previous steps (Fig. 1d). Finally, the routability-check step verifies whether all control flows can be scheduled or not, given (bandwidth and latency) requirements [1], [2]. Input to the framework are the topology description, traffic patterns in the network and a set of optimization parameters. The framework output consists of the mapping, the association, a table specifying possible controller flows in the topology (similar to an SDN flow table) and the result of the optimization parameters (allocated bandwidth, bounded latency and minimum reliability). Essentially, the approach presented in [1]–[3] is designed to quickly produce congestion-free deployment plans following reliability, bandwidth and latency requirements. Note that reliability here refers to service availability, i.e., the minimum probability that the nodes are operational and connected to at least one controller. The approach can also be used to systematically evaluate and analyze the consequences of a deployment plan with respect to the requirements, e.g., for the purpose of network planning [2].

III. SIMULATION TEST-BED IMPLEMENTATION

The overall implementation of the simulation test-bed and its interaction with the optimization framework are presented...
Fig. 2. Implementation of the simulation test-bed using NS-3 [4].

in Fig. 2. TopologyReader provides the network topological information to the SimulationBuilder. ResultsParser reads the optimization framework output and delivers it to SimulationBuilder, which has two components: one for storing and handling the information (InfoHandler) and one for building and executing the ns-3 simulation (NS3-CompBuilder). ns-3 functionality was extended by implementing Weighted-Fair-Queuing (WFQ) and a TrafficGenerator, where traffic flows are modeled as UDP socket applications. TrafficGenerator generates packets based on burstiness-demand pairs delivered in FlowsInfo based on a leaky-bucket approach [5]. Inter-packet transmission time is drawn from an exponential distribution, reflecting the expected packet rate, combined with a Bernoulli distribution controlling the burstiness. Finally, the SimulationBuilder delivers the simulation results to the Visualizer for display purposes.

IV. DEMO

We demonstrate the efficiency of the DCP approach over topologies of varying size through two scenarios, comprising: 1) fast (re-)calculation of deployment plans under changing traffic conditions, and 2) resilience and fast mitigation under node and link failures. Fig. 3 outlines the main steps of the demo, here exemplified by the use of the DataXchange topology [6] chosen for visualization simplicity. Note that the framework scales well with significantly larger networks—in [1] we showed that a deployment plan for the larger InternetMCI [6] topology can be retrieved within around 22s, which is 80x faster than when using CPLEX. A provisional video of the demo can be found online at https://bit.ly/36hnOWc.

In the first scenario (Fig. 3a and b), we demonstrate the case of transitioning from low-traffic network conditions (total control-traffic demand of 0.9 Mbit/s) in early mornings to high-traffic during e.g. peak hours (total control-traffic demand of 11 Mbit/s). A new CP solution better suited for peak flow demands is generated within seconds (e.g. 3s for DataXchange) given original reliability, bandwidth and delay requirements. The new plan is employed to quickly adjust control-traffic flows routes in line with recalculated bandwidth allocations, achieved minimum reliability and estimated worst-case end-to-end delays (Fig. 3b). In both cases, performance guarantees are maintained in line with the actual network conditions and traffic demands.

In the second scenario, we demonstrate that the approach provides a CP deployment that is highly robust to link and/or node failures (Fig. 3c and d). Ensuring controller service availability even under network failures is crucial in, e.g., emergency situations to facilitate communication between societal emergency services. Given traffic flow demands, we show that the deployed controller plane is capable of maintaining inter-controller and controller-aggregator connectivity while re-calculating an updated deployment plan, which can handle the original traffic demands under given performance requirements over the nodes and links that are still operational.

V. CONCLUDING REMARKS

The showcased optimization framework enables unprecedented fast computation of DCP deployments while accounting for performance requirements, which previously took prohibitively long time to process for practical applications. The approach is proactive: it accounts for routability requirements instead of reacting to failures and traffic congestion. This is vital for current and future time-critical services integration. These characteristics enable a truly elastic DCP adaption to changing network conditions for effective resource utilization. Additionally, the approach may serve as a tool for quantifying the trade-off between bandwidth, delay and reliability.

ACKNOWLEDGMENT

This work has been financially supported by the Swedish Foundation for Strategic Research (SSF) Time Critical Clouds (grant no. RIT15-0075) and Celtic Plus 5G-PERFECTA (Vinnova, grant no. 2018-00735).

REFERENCES


