

Paper Review November 12th

Jiaqi Yang (jxy530)

Decentralized Consistent Updates in SDN

What problem is the paper solving and why is it important?

The problem that this paper addresses and tries to solve is that updating data plane state to dynamic condition by performing updates as quickly as possible while preserving certain consistency properties is a common challenge faced in many software-defined network (SDN) systems. Since updating data plane is a fundamental and essential operation in all centrally-controlled networks, the speed is crucial: performing updates as fast as possible is paramount in a variety of scenarios ranging from performance, fault tolerance, to security. To infinitely approach the ideal case, which is updating its network-wide state while preserving consistency instantaneously, the authors of this paper proposes ez-Segway, a decentralized mechanism to consistently and quickly update the network state while preventing forwarding anomalies and avoiding link congestion.

What is the main idea of the paper?

The main insight of this paper is that by proposing ez-Segway, a new mechanism for network updates. The key insight is to involve the switches as active participants in achieving fast and consistent updates. The controller is responsible for computing the intended network configuration and identifying flow segments and dependencies among segments. The update is then realized in a decentralized fashion by the switches: they execute a network update by scheduling update operations based on the information received by the controller and messages passed among neighboring switches. With this philosophy in mind, this new idea allows every switch to update its local forwarding rules as soon as the update dependencies are met without any need to coordinate with the controller.

How does the paper differ from previous work?

Although there are handful of researches and developments to address the problem of performing updates as quickly as possible while preserving certain consistency properties, the bulk of previous work in this area focused on maintaining consistency properties: since updates cannot be applied at the exact same instant at all switches. However, all these approaches require multiple rounds of communications between the controller and the switches. Therefore, all the previous works has certain drawbacks: 1) the update time is inflated by inherent delays affecting communication between controller and switches, 2) the update time is slowed down by a centralized computation, 3) the update process cannot quickly adapt to current data plane conditions, and 4) updates require additional coordination overhead among different controller. However, ez-Segway with a decentralized fashion only does the pre-computation and solves all the problems previously mentioned.

Are there any flaws in the paper? How would you improve the paper or build on it in future work?

Although it is really attractive to loosen the burden of computation and arrangement from the controller, the new proposed mechanism, ez-Segway, makes the switches as active participants in achieving fast and consistent updates. In this way, this new idea allows every switch to update its local forwarding rules as soon as the update dependencies are met without any need to coordinate with the controller. However, this corresponds to higher requirement on the hardware of the switches since now they have to share part of computation workload, updating, and conserving the consistency. The hardware requirement is not mentioned in the article, therefore this could be potential drawback of this approach.