Hierarchical Controller Architecture in Software Defined Networks

{ Gourav Khaneja, Sweta Seethamraju & Praveen Kumar }

Problem

Typical Control plane architecture in SDN involves multiple peer controllers, managing subset of network, but keeping a global view of the network.

- Replication of entire network state requires huge communication overhead: hinders scalability of the system
- Network inconsistency degrades the performance of control application
- Scalability of global network view itself

Proposed Solution

Switch Abstraction

- A controller creates an abstract switch representing the underlying managed network.
- A controller exposes "OpenFlow" API to manage the abstract switch by other (parent) controller.
- OpenFlow messages are translated between parent controller and underlying network.

Hierarchical arrangement

- Controllers are arranged in a tree structure over the network using switch abstraction.
- Controllers do not keep a global view of network, but only the part which they control.
- Local application and network events are shielded from upper layer controllers.

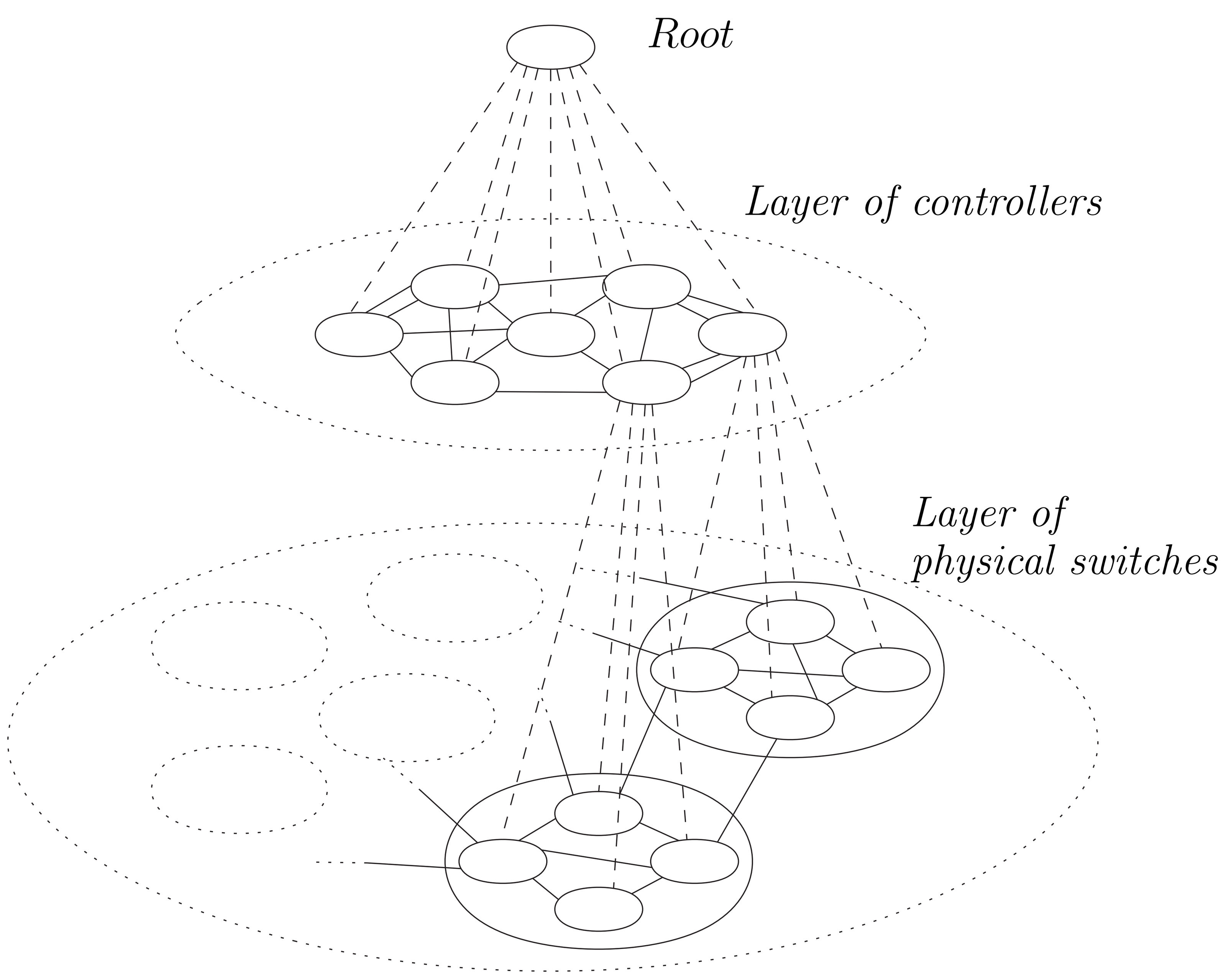

Figure 1: Hierarchical arrangement of controllers

Pros

- Reduces replication overhead
- Minimizes network inconsistency
- Enables smooth transition across management barriers

Price we pay

Sub-optimality for inter-domain operations. How much is the deviation from optimality ?

Experimental Setup

- Flow data (6000 MapReduce jobs) was generated by sampling historical Hadoop traces on a 600 machine cluster at Facebook (1 million jobs). A dummy MapReduce application choses directory, map and reduce nodes randomly for each job.
- Topology: 600 hosts connected through a network of 200 switches and 10 Gbps links. The cluster is divided into 10 equally-sized control domains.
- TE-cum-routing control application: When a new flow arrives, it finds the path with minimum maximum utilization. If multiple such paths exist (which is frequently the case, due to a large network), shortest path (in term of number of hops) is chosen.

Metrics

- Path length (number of hops) of the routes assigned to flows.
- Maximum link utilization of the path assigned to flows.
- Communication overhead (number of messages exchanged between controllers) for setting up routes for flows.

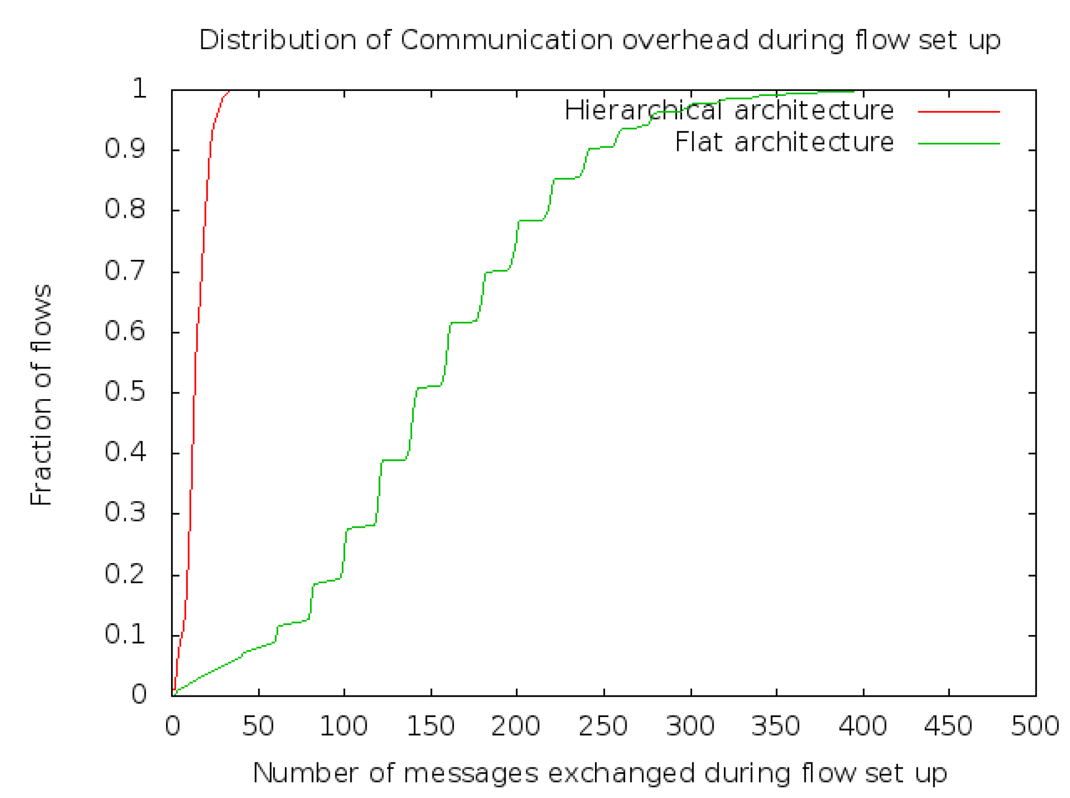

Figure 2: Per flow communication overhead

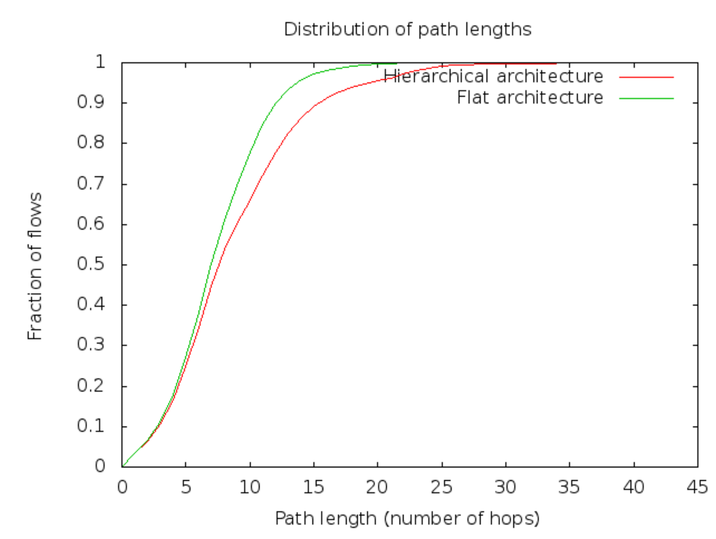

Figure 3: Path length

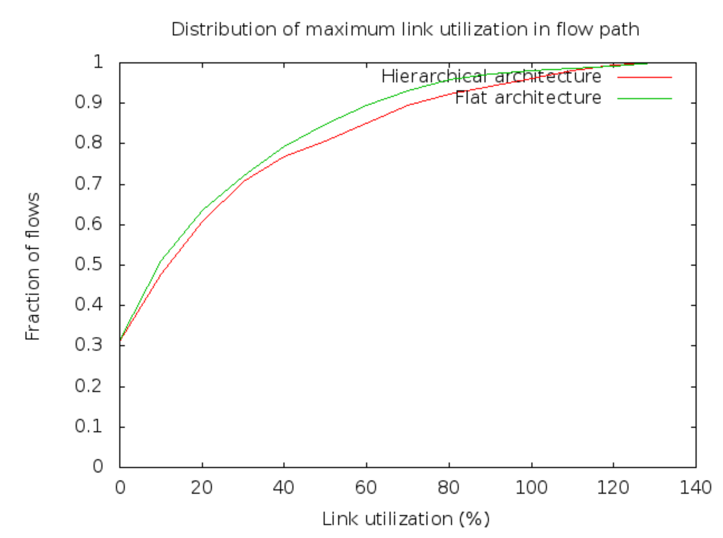

Figure 4: Maximum link utilization

Results

Hierarchical architecture reduces communication overhead (Figure 2) by an order of magnitude while compromising modestly on optimality: shortest path length (17.4%) (Figure 3) and minimum maximum link utilization (12.8%) (Figure 4).

Resources

https://github.com/MugiwaraLuffy/ACN/
https://github.com/SWIMProjectUCB/SWIM