Capacitated Next Controller Placement in Software Defined Networks

Bala Prakasa Rao Killi and Seela Veerabhadreswara Rao

IEEE Transactions on Network and Service Management - Sep. 2017

Presented by: Rafael Lourenço

Networks Lab - UC Davis

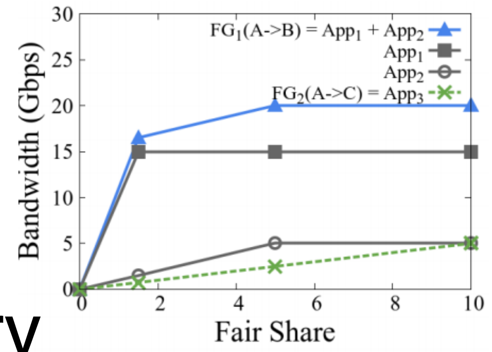
November 10, 2017

UCDAVIS
UNIVERSITY OF CALIFORNIA

Software Defined Networking Control Plane

- Using a single controller is beneficial as it provides centralized management. However:
  - Increases latency of switches that are far from controller
  - Bottleneck in processing
  - Single point of failure for entire network
- Therefore: control plane logically centralized, but physically distributed across multiple controllers
  - All controllers maintain consistent global view to operate the network
  - Controllers deployed at specific locations for optimal performance
  - Each switch receives forwarding rules from a unique controller at once (while assigned to multiple controllers for reliability)

How Traffic Engineering (TE) works in B4?



1. Find up to 4 end-to-end paths (*tunnel group*) for every demand (src-dst-class_of_service)
2. Use a Waterfill algorithm to allocate bandwidth to demands, starting at the cheapest path (*tunnel*) for each demand. The goal of this step is to find weights for different tunnels:
    1. Equally fill the demand's utility functions until a bottleneck link in that tunnel is reached
    2. When a bottleneck is reached, freeze all flows whose tunnels go through said link and proceed to next tunnels
    3. When process is done, every demand will be allocated to a tunnel group and every tunnel in that group will be assigned a weight
3. The weights found before are quantized to multiples of 0.25

UCDAVIS

[1] B4: Experience with a globally-deployed software defined WAN, SIGCOMM 2013

Controller Placement Problem

- Generally: determining the optimal location of controllers and the assignment of switches to them is known as the Controller Placement Problem

➢ B. Heller et al., "The controller placement problem," *Hot Topics in Software Defined Networks*, 2012:
  - ➢ Metrics such as worst case and average case latency between switches and controllers; maximizing number of switches within a delay bound
  - ➢ Analyzed impact of number of controllers and their placement on latency, and the choice of metric on placement
  - ➢ **Did not consider the capacity of controllers**

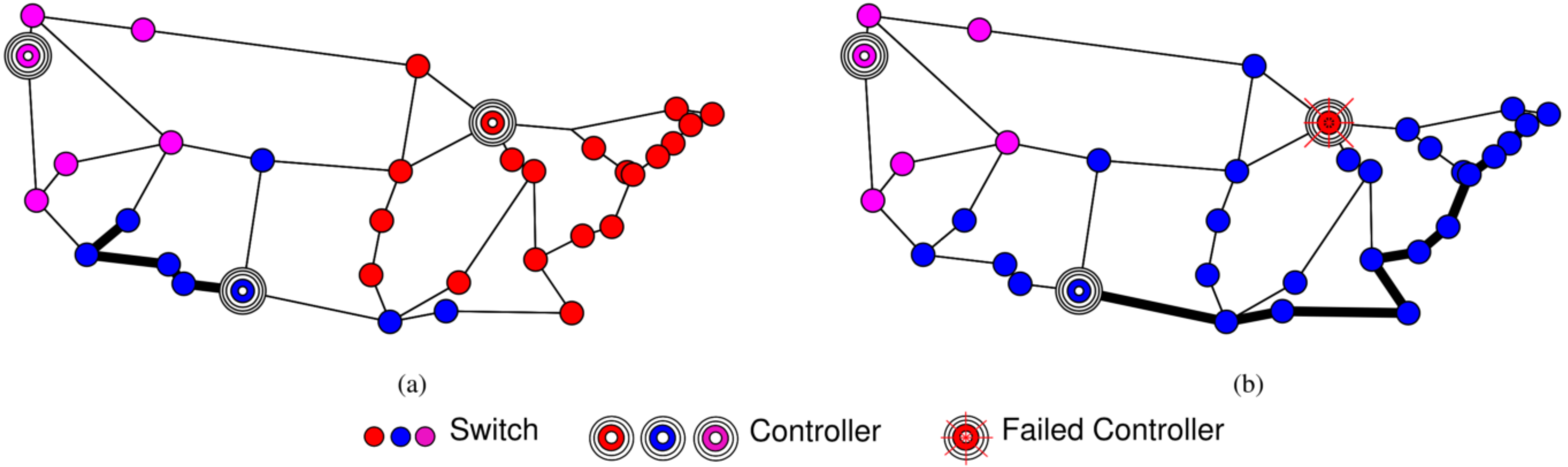
UCDAVIS

Capacitated Controller Placement Problem

- G. Yao et al., “On the capacitated controller placement problem in software defined networks,” *IEEE Communication Letters,* 2014:
  - Incorporated a constraint on controller capacity into CP
  - Objective: minimize the worst case latency while satisfying the capacity constrains.
  - **Does not take reliability into account while deploying controllers**

UCDAVIS

Impact on Latency of Controller Failures



- Worst-case latency from switch to controller (bold paths) without pre-computed secondary controller (i.e., using controller assignment ”restoration”, and not ”protection”):
  - No failure = 7.68ms
  - With failure = 19.66ms

Capacitated **Next** Controller Problem

- Considers capacity and reliability of controllers
- Also plans ahead for controller failures
- I.e.: given the number of controllers to be deployed, determine simultaneously the location of controllers and the assignment of switches to them
- Objective: minimize the maximum, for all switches, of the sum of the latency from the switch to the nearest controller with enough capacity (*first reference controller*) and the latency from the first reference controller to its closest controller with enough capacity (*second reference controller*)

UCDAVIS

CNCP MILP Formulation

- (Next slide) For first and second reference controller, i.e. single controller failure preparedness
- Minimizes the worst-case latency in case of controller failure, with only double-indexed variables
- To minimize average latency, necessary to include three-indexed variable
- Authors also present a general formulation for multiple failures (not in these slides)

$$\sum_{j \in P} y_j = p \tag{12}$$

$$\sum_{j \in P} r^1_{ij} = 1 \quad \forall i \in S \tag{13}$$

$$\sum_{\substack{j \in P \\ j \neq i}} r^2_{ij} = 1 \quad \forall i \in S \tag{14}$$

$$r^1_{ij} + r^2_{ij} \leq y_j \quad \forall i \in S, \forall j \in P \tag{15}$$

$$w_{jk} d_{jk} \leq \gamma G_d \quad \forall j,k \in P \tag{16}$$

$$w_{jk} \geq y_j + y_k - 1 \quad \forall j,k \in P \tag{17}$$

$$w_{jk} \leq y_j \quad \forall j,k \in P \tag{18}$$

$$w_{jk} \leq y_k \quad \forall j,k \in P \tag{19}$$

$$y_j + \sum_{\substack{h \in P \\ d_{ih} > d_{ij}}} r^1_{ih} \leq 1 \quad \forall i \in S, \forall j \in P \tag{20}$$

$$\sum_{i \in S} L_i r^1_{ij} + \sum_{i \in S} L_i r^2_{ij} \leq U_j y_j \quad \forall j \in P \tag{21}$$

$$z \geq (d_{ij} + d_{jk})\left(r^1_{ij} + r^2_{ik} - 1\right) \quad \forall i \in S, j,k \in P \tag{22}$$

Variable	Description
	0, otherwise
	=1, if j is the second reference controller of switch i;

12: Exactly *p* controllers are deployed in the network

13 & 14: Each switch has unique first and second reference controllers respectively

14: Second reference controller of switch *i* must be different from the one that is deployed at *i*

15: *j* is either the first or the second reference controller to a switch *i* (i.e., *i* and *j* must be different)

16: Latency between any pair of active controllers is less than γ**Gd*, i.e., the maximum allowable inter controller latency

17 to 19: Avoid the non linear term (*Wjk*) in the formulation and make it linear

20: *Closest assignment constraint* – switches are assigned primarily to the closest controller, and, in case of failures, only then to the other controller

21: total demand of the switches served by a controller *j* does not exceed it's capacity *Uj*

22: objective value greater than sum of latency from switch *i* to its nearest controller *i* and the latency from *i* to its nearest controller *j*

Controller Failover

- Presented formulation generates the initial controller locations and the switch to controller assignment as output, and also generates all the *Q*-reference (first, seconds, etc) controllers for every switch
- Master controllers receive PACKET_IN messages, as it fails, methods [1] to [5] can be used to make the following Qth controller the master

[1] N. Katta et al., "Ravana: Controller fault-tolerance in software-defined networking," *SIGCOMM 2015*
[2] W. Chen et al., "On the quality of service of failure detectors," *IEEE Trans. Comput.* 2002.
[3] N. Hayashibara et al., "The ϕ accrual failure detector," *IEEE Symp. Rel. Distrib. Syst. 2004*
[4] B. Satzger et al., "A new adaptive accrual failure detector for dependable distributed systems," *ACM S. Appl. Comput. 2007*
[5] T.-W. Yang and K. Wang, "Failure detection service with low mistake rates for SDN controllers," *APNOMS 2016*

Heuristic Solution

- Since the simplest formulation has a quadratic number of variables, and the average case and multiple failure formulations have cubic number of variables a heuristic is proposed
- Heuristic uses simulated annealing by finding ”neighbor” solutions, evaluating the objective function (minimize worst-case post-failure latency, e.g.), keeping the best with some probability and keeping the worse with the complement probability (probability decreases with temperature, from T_max to T_min – user inputs)

UCDAVIS

Results

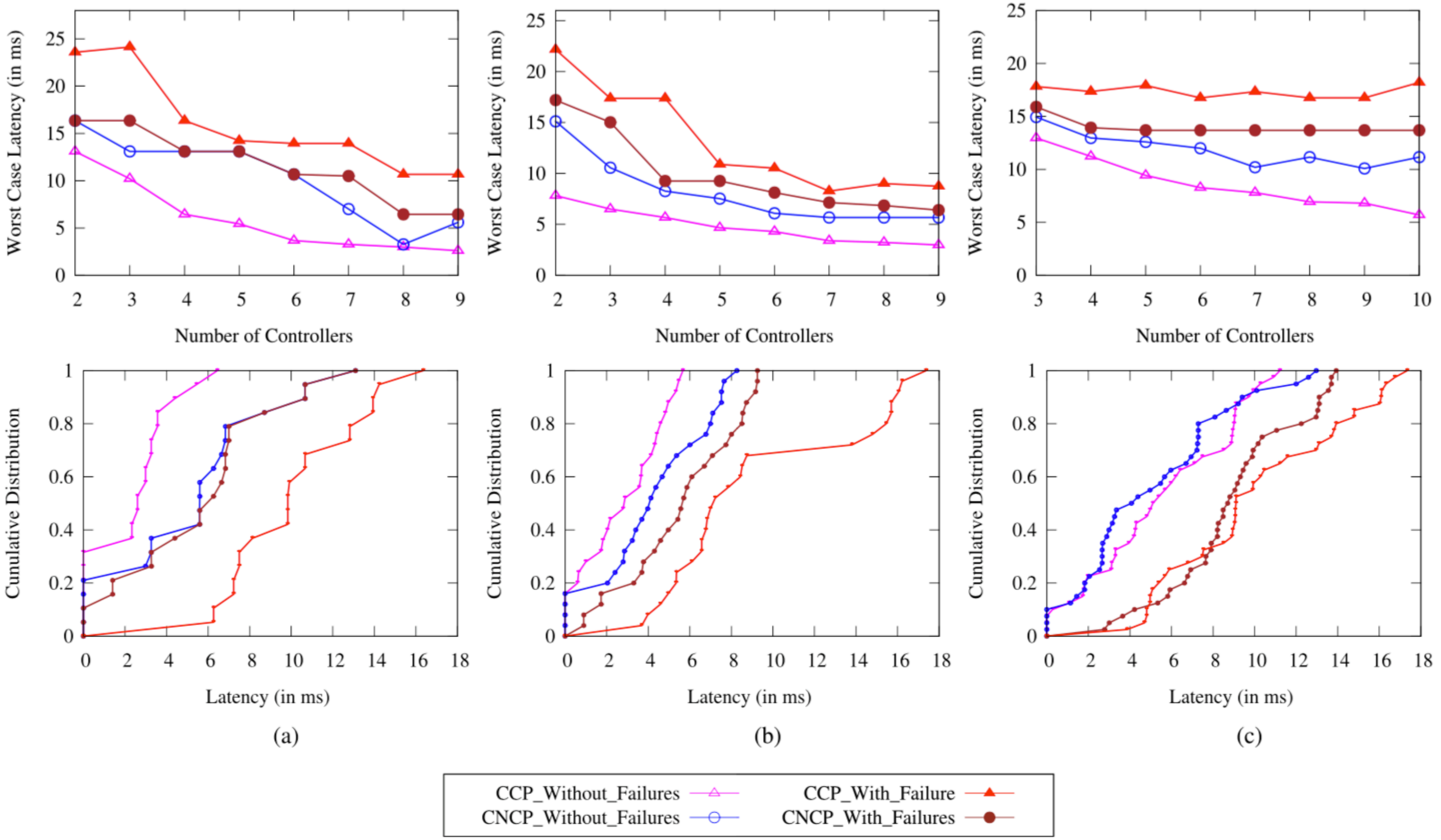
- Authors use three Internet Topology Zoo topologies: AARNET (19 nodes), AT&T (25 nodes), and GEANT (40 nodes)
- Three metrics:
  1. Worst-case latency (no failure scenarios):
     a) For Capacitated Controller Placement Problem (CCP) is the maximum, for all switches, of the latency from the switch to its nearest controller
     b) For Capacitated Next Controller Placement Problem (CNCP) is the maximum, for all switches, of the latency from the switch to its nearest controller summed with the latency from that controller to the next controller closest to it
  2. Maximum worst-case latency: defined in is the maximum, for all possible failure scenarios, of the worst-case latencies
  3. Inter-controller Latency

UCDAVIS

Single Failure

Worst-case and Maximum Worst-Case Latencies

Cumulative Latencies



- 1 controller ~ 30 switches (for CNCP)

UCDAVIS

Single Failure

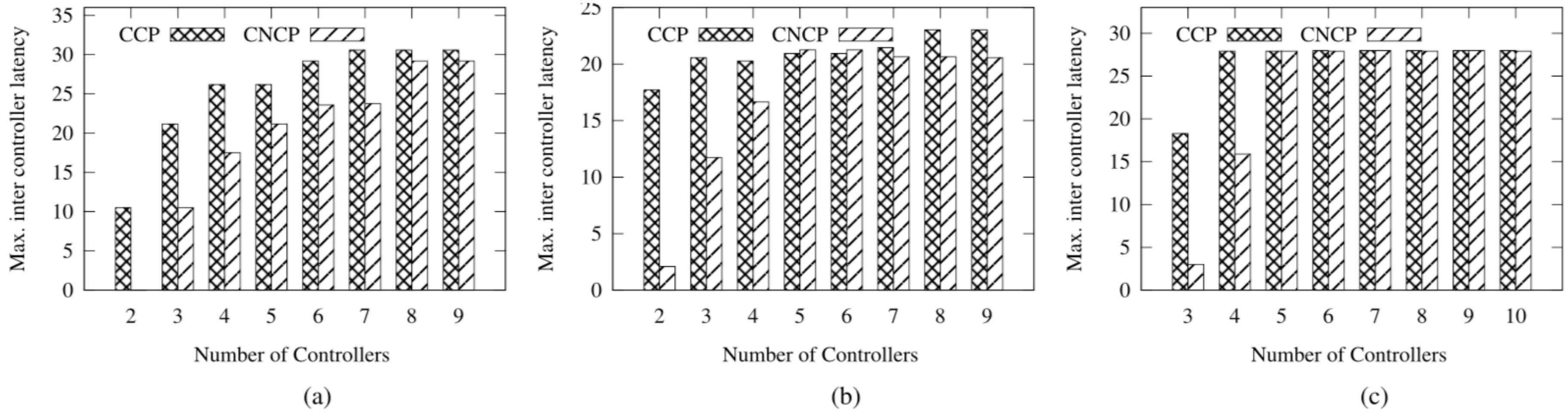


Fig. 4. Maximum inter controller latency of CCP and CNCP on various networks (a) AARNET (19 Nodes). (b) AT&T (25 Nodes). (c) GEANT (40 Nodes).

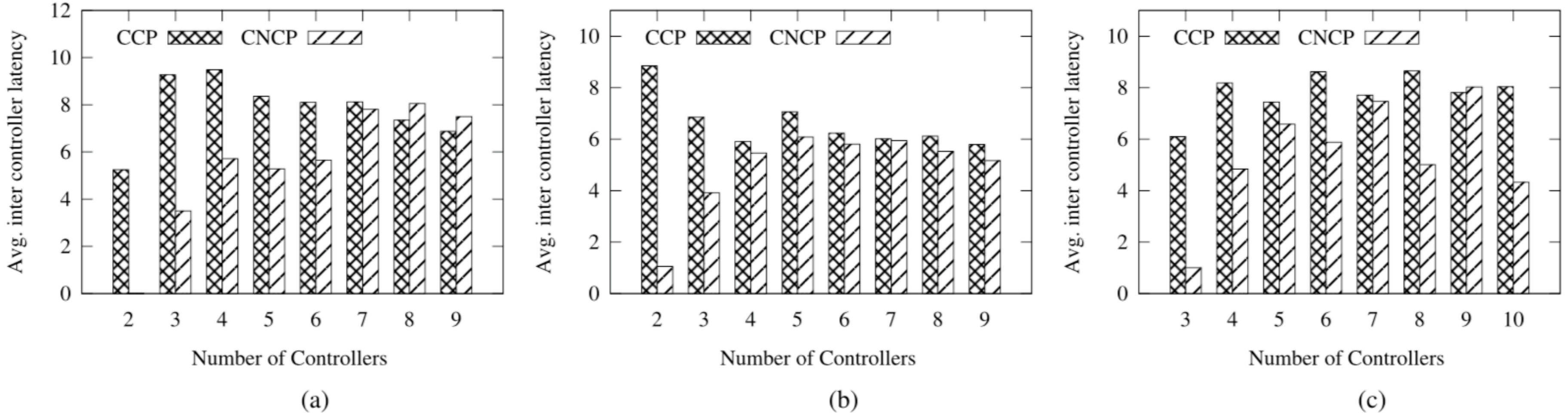


Fig. 5. Average inter controller latency of CCP and CNCP on various networks (a) AARNET (19 Nodes). (b) AT&T (25 Nodes). (c) GEANT (40 Nodes).

Double Controller Failure

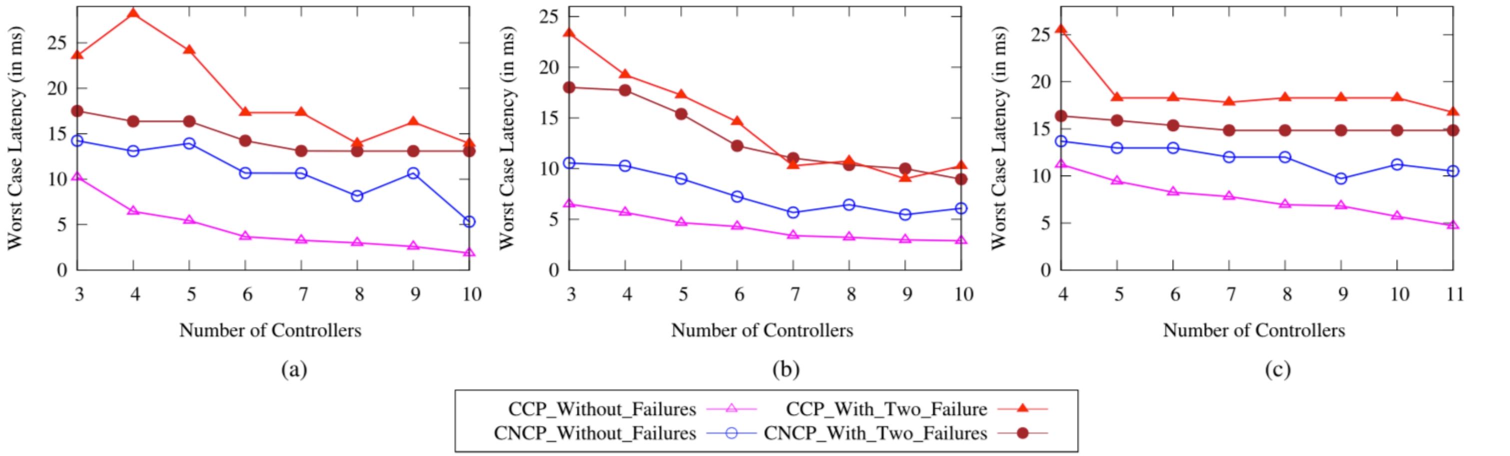


Fig. 9. Worst case latency and maximum worst case latency of CCP and CNCP while evaluating for two controller failures (a) AARNET (19 Nodes). (b) AT&T (25 Nodes). (c) GEANT (40 Nodes).

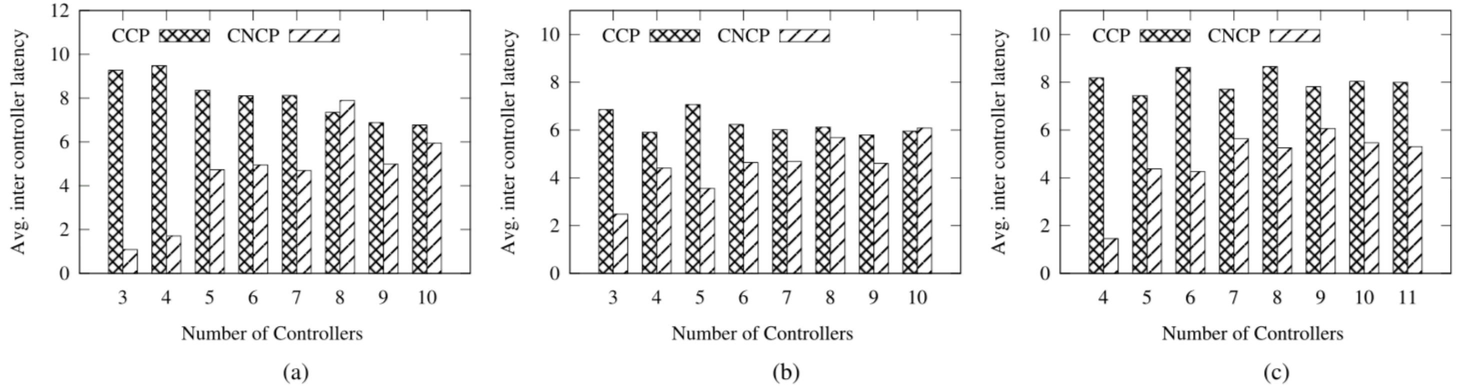


Fig. 11. Average inter controller latency of CCP and CNCP while evaluating for two controller failures (a) AARNET (19 Nodes). (b) AT&T (25 Nodes). (c) GEANT (40 Nodes).

Takeaways

- Authors present the idea (although not referring to it) of protection versus restoration in terms of SDN controllers
- Interesting MILP formulation based on Next P-center Problem [1], however:
  - Formulation forces first and second controllers (i.e., main and secondary controllers) of a switch to be close to each other: In case of disaster, both have higher chance of being affected and, thus, making the proposed solution extremely dangerous for geographically correlated failures
  - Always shortest paths (lest latency-impacting) between controllers and between controllers and switches

UCDAVIS