

A decorative background featuring a network diagram with nodes and connecting lines, primarily located in the top right and bottom left corners.

基於SDN的資料中心大象流負載平衡

Load Balancing for Elephant Flows in SDN-based Data Center

指導教授：蔡孟勳

組別：3-9組

專題成員：尹子維

Outline

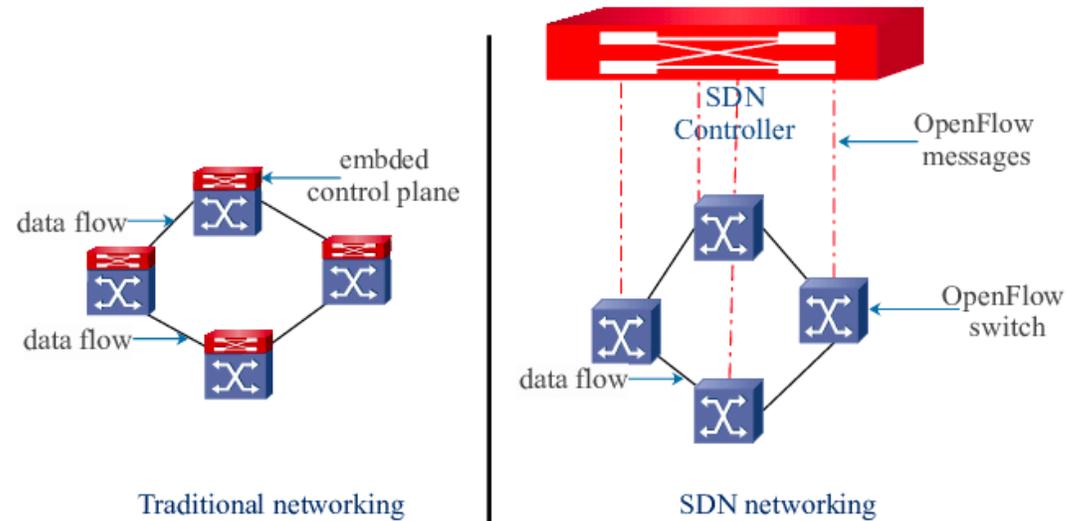
- ◎ 簡介
- ◎ 方法介紹
- ◎ 實驗結果



簡介

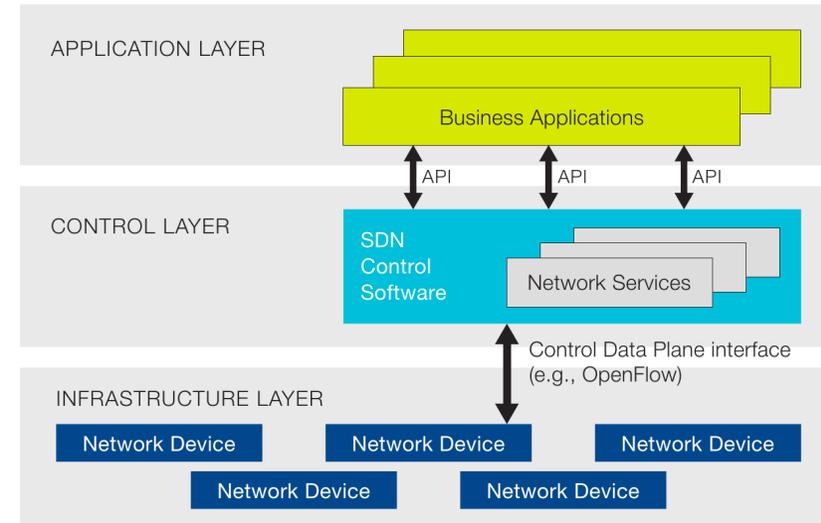
SDN

◎ SDN有別於過去網路，將控制層和資料轉發層分離。



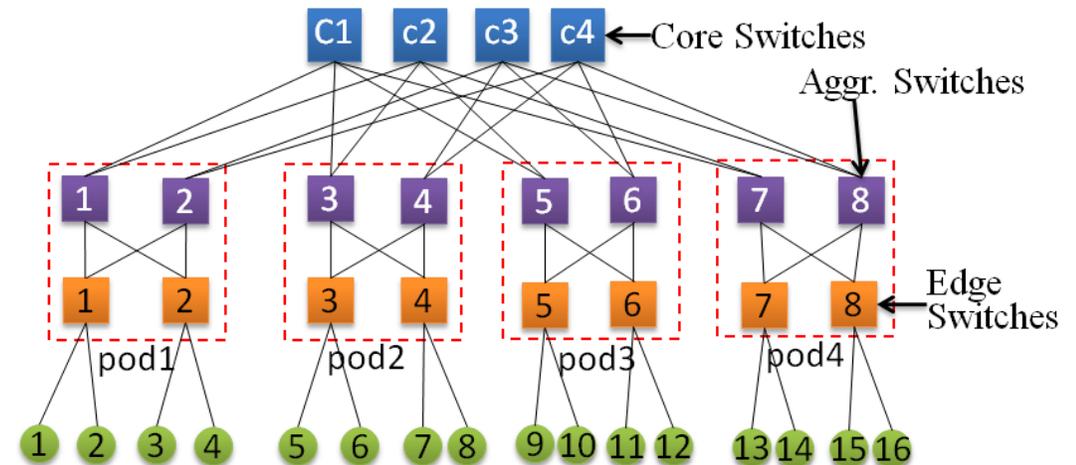
OpenFlow and Ryu

- ◎ OpenFlow提供了分離後的控制層和資料轉發層之間的通訊協定。
- ◎ Ryu提供了實用的控制層的API，也提供了和外部程式連結的北向介面。



K-ary Fattree

- ◎ K-ary Fattree是常見在資料中心使用的網路拓撲，分為Core switch, Aggregate switch和Edge switch。
- ◎ Core switch僅會和Aggregate switch相連
- ◎ Aggregate switch和Edge switch會被分成一個一個的pod，並且只有同一個pod中的交換器會相連。
- ◎ 每一個交換器會有k個port



研究問題

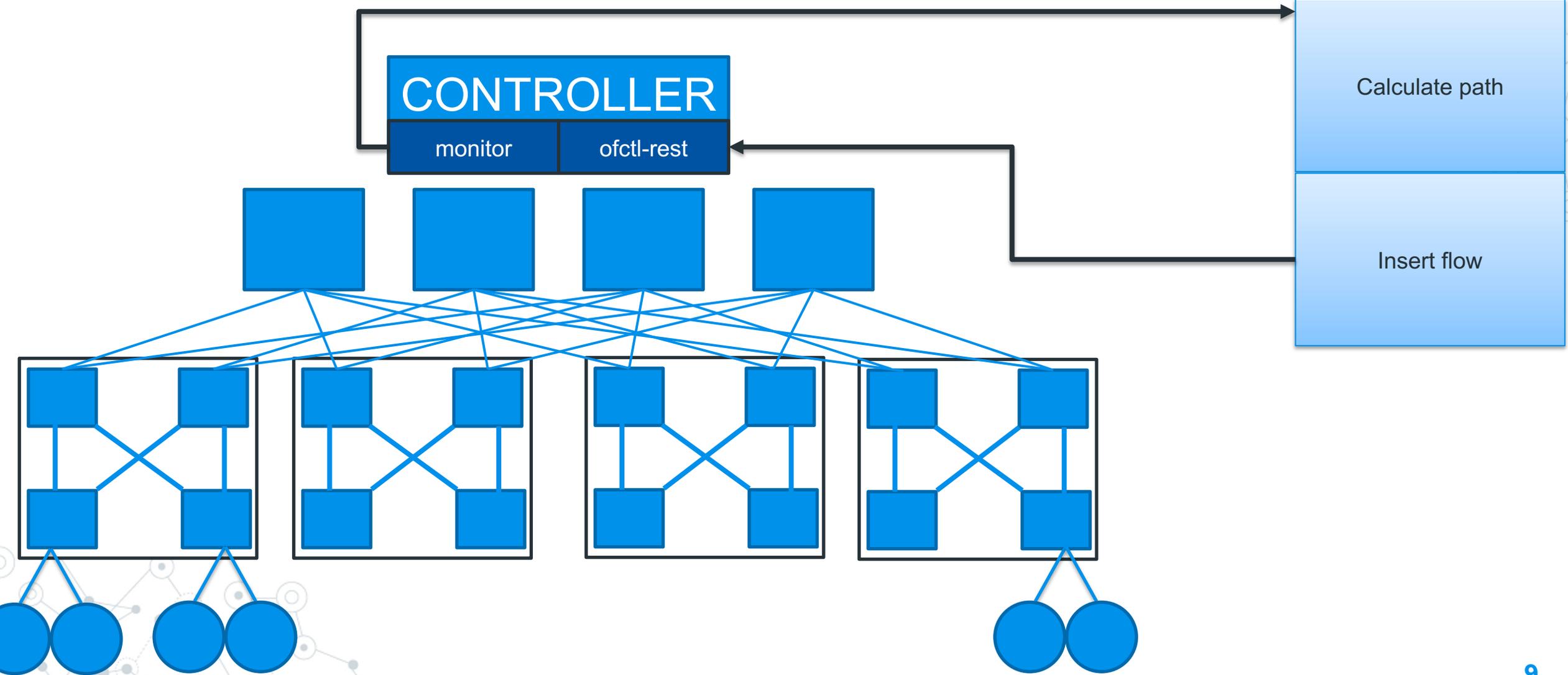
- ◎ 在SDN網路中，經常會以更新Flow table的方式來實現負載平衡。
- ◎ 在過去的研究中，有人提出以Dijkstra's Algorithm來計算最短路徑，並讓Controller來更新switch中的flow table來實現負載平衡。
- ◎ 在另一個研究中，針對大象流，他提出讓大象流分流來實現負載平衡，此篇中判斷大象流的方式是以給定的閾值來做判斷。
- ◎ 若網路中壅塞程度較低時，某些較大的流量理應不用特地做分流，可以使用較為快速的Dijkstra's Algorithm來實現load balance。
- ◎ 因此我希望讓判斷大象流的標準是以網路整體狀況來看待，這樣在做load balance時可以選用更加適合的算法。

A decorative network diagram in the top-left corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow with a grey outline. The connections are a mix of solid and dashed lines, creating a complex web-like structure.

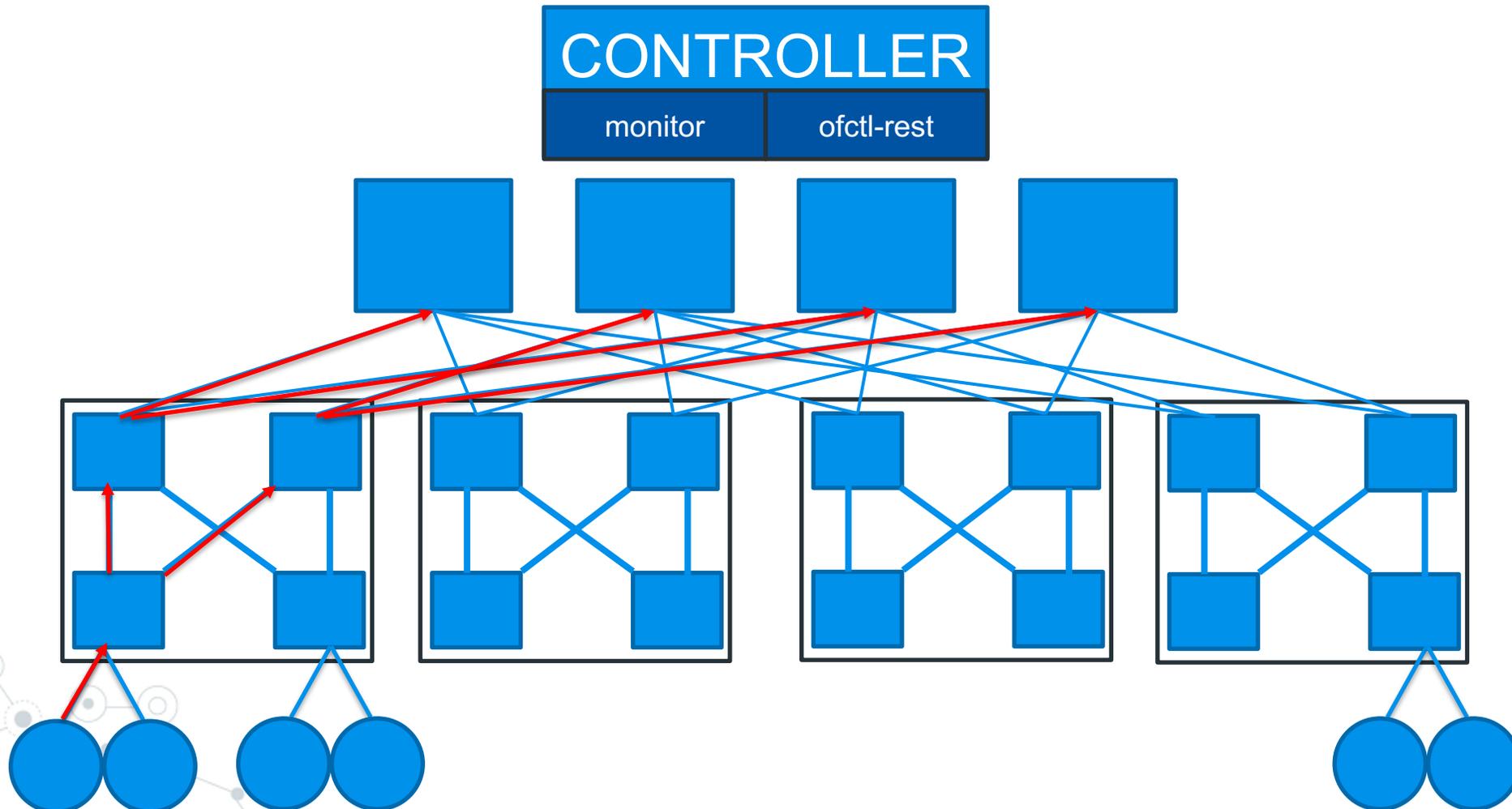
方法介紹

A decorative network diagram in the bottom-right corner, similar to the one in the top-left. It features a cluster of nodes connected by lines, with some nodes being solid grey and others hollow with grey outlines. The overall style is clean and modern, typical of a technical or scientific presentation.

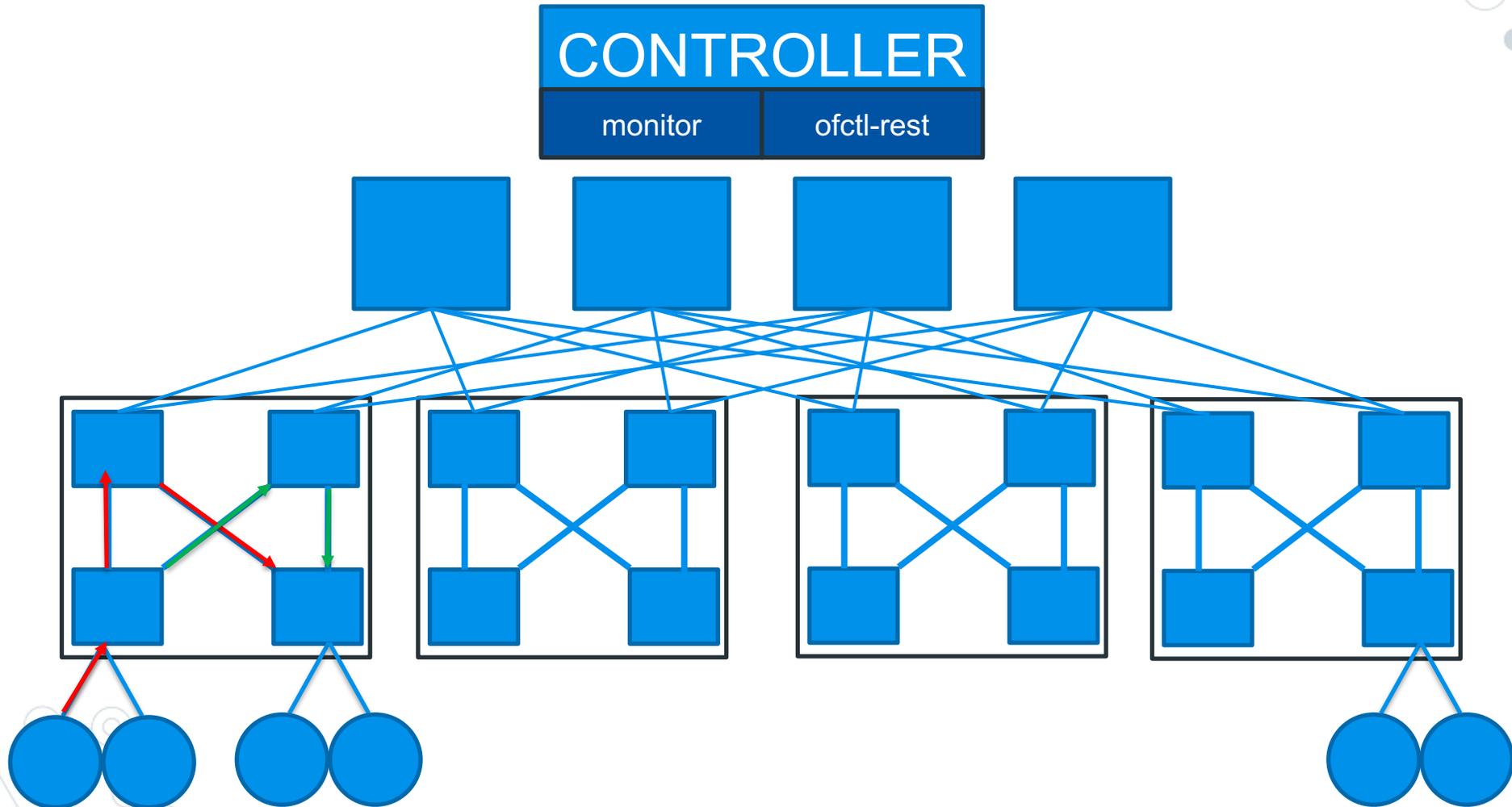
程式架構



Split Flow Algorithm



Update Flow By Dijkstra's Algorithm



Elephant Flow Decide

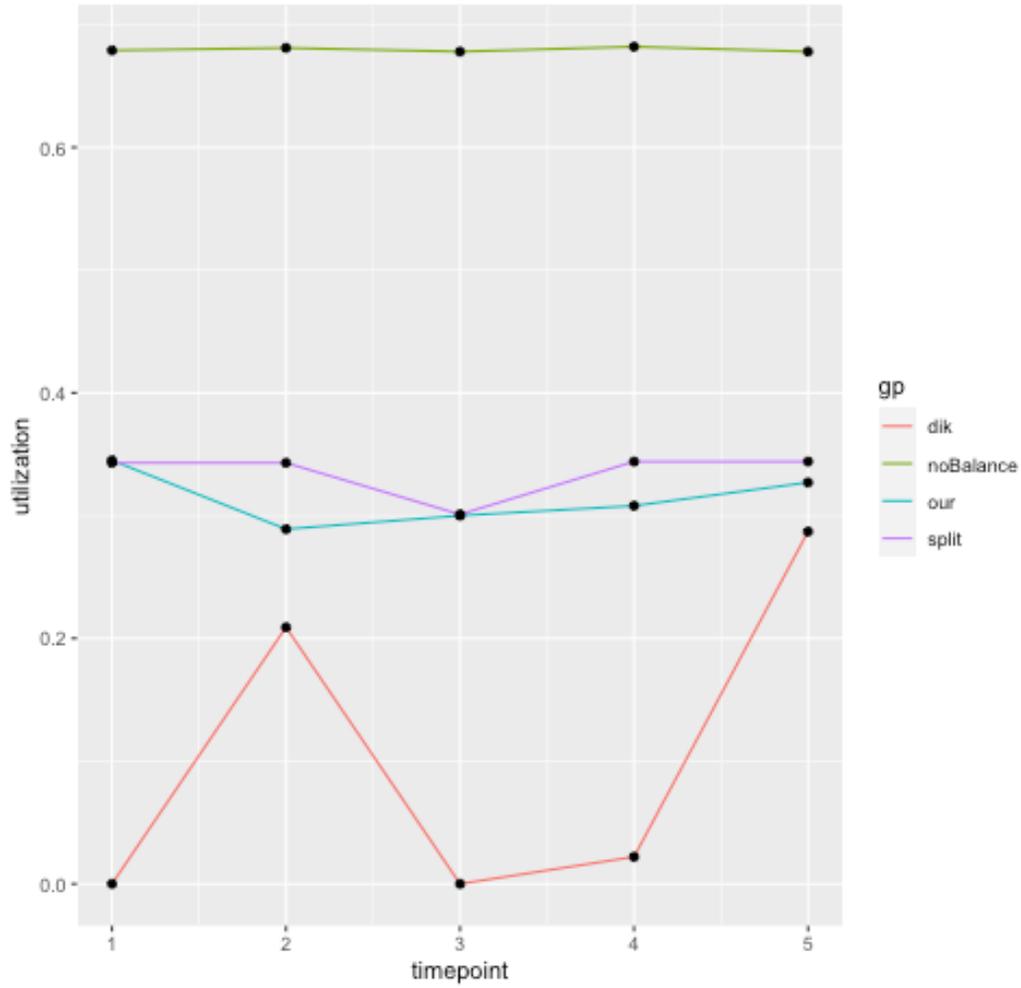
1. L_i : link i .
2. Fit a beta distribution for L_1, L_2, L_3, \dots
3. A beta distribution has two parameters $\hat{\alpha}, \hat{\beta}$
4. Mean : Use data mean
5. Median $\approx \frac{\hat{\alpha} - \frac{1}{3}}{\hat{\alpha} + \hat{\beta} - \frac{2}{3}}$
6. If mean \geq median \Rightarrow use elephant flow split flow algorithm
7. If mean $<$ median \Rightarrow use Dijkstra shortest path



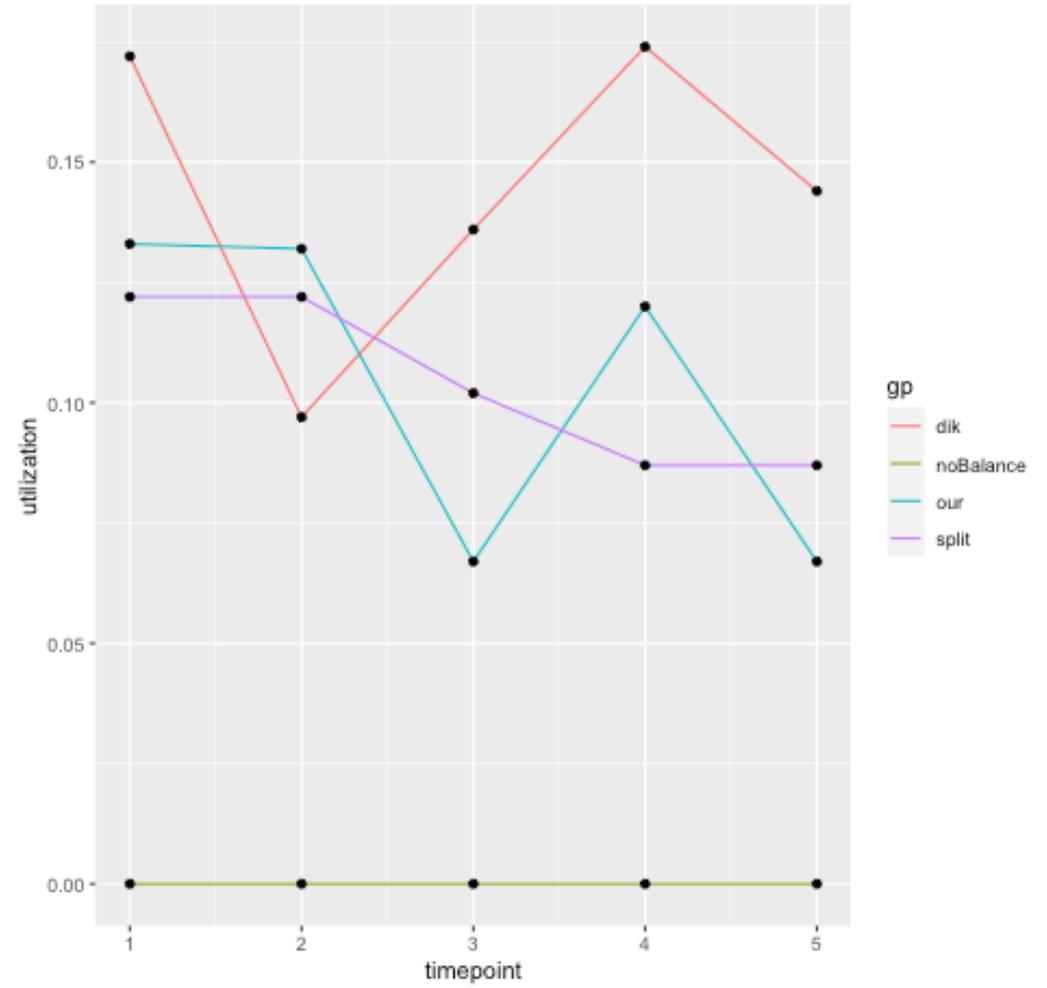
實驗結果

Core To Aggregate

Core To Aggr Link1

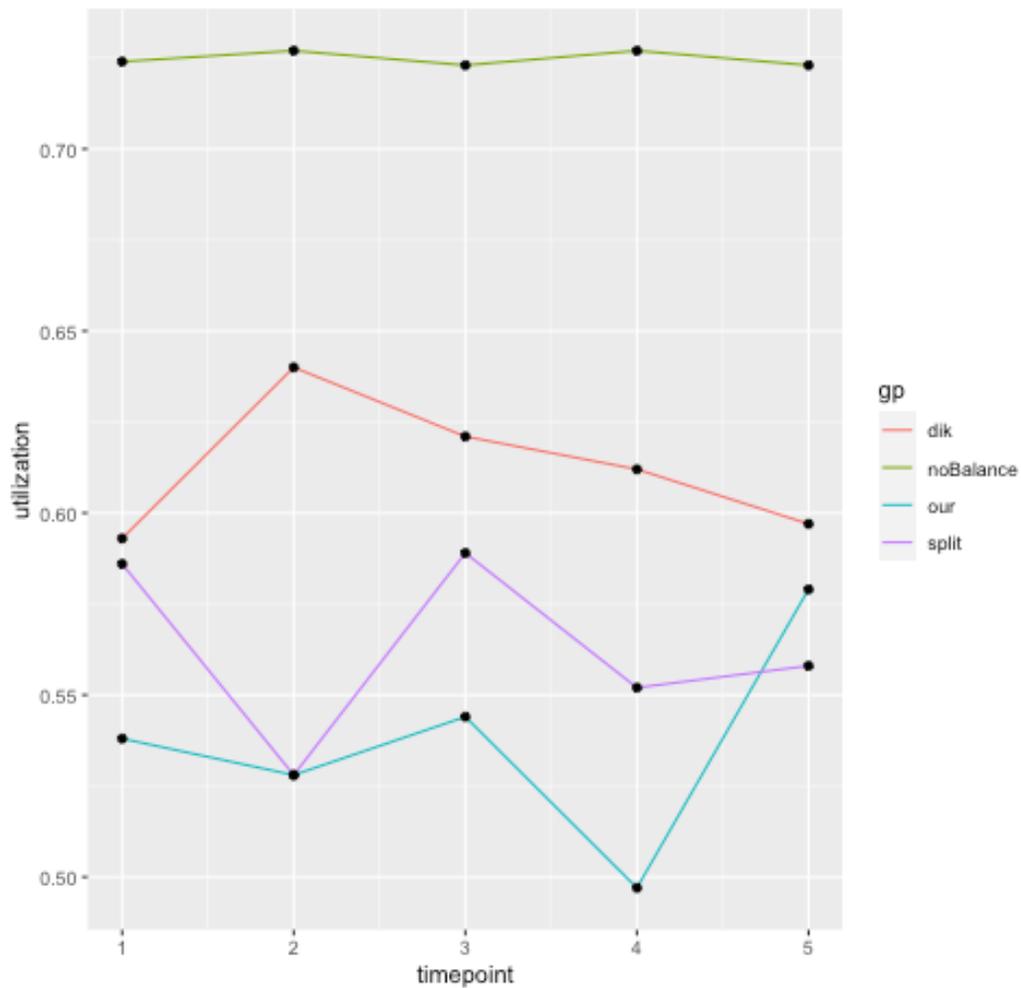


Core To Aggr Link2

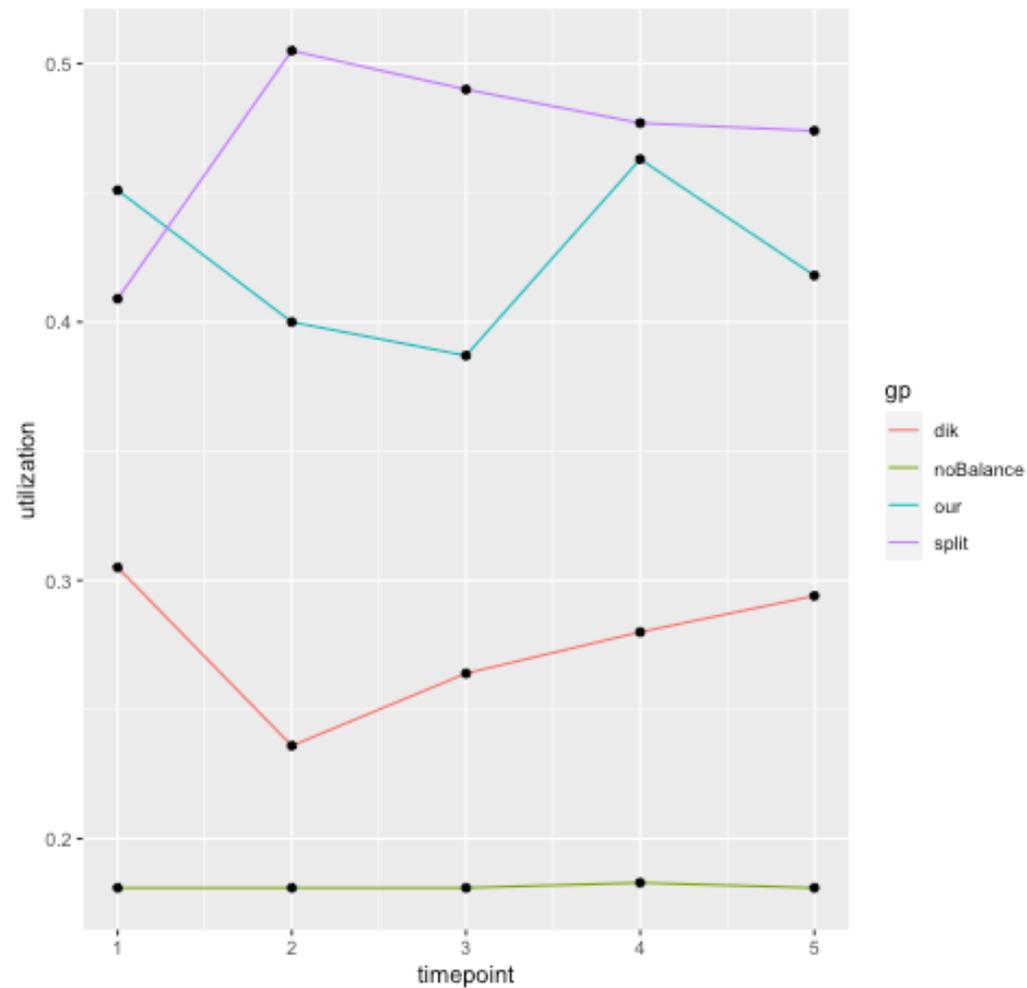


Aggregate To Edge

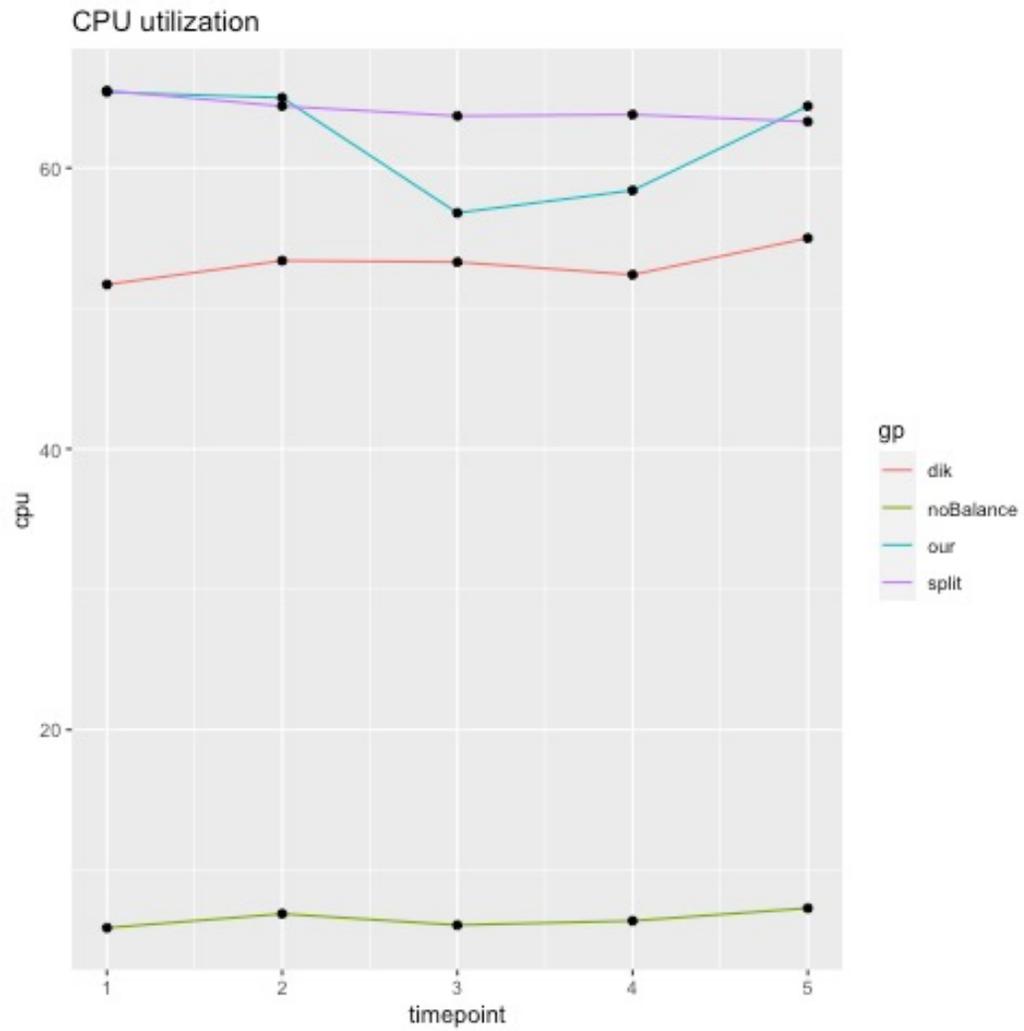
Aggr To Edge Link1



Aggr To Edge Link2



CPU utilization



A decorative network diagram in the top-left corner, consisting of various sized grey circles connected by thin grey lines, some solid and some dashed, forming a complex web structure.

THANKS!

A decorative network diagram in the bottom-right corner, similar to the one in the top-left, featuring grey circles of different sizes connected by thin grey lines, some solid and some dashed, creating a web-like pattern.