Open Research Objects (ORO)

Research Objects Reviewed (ROR)

Control Channel Isolation in SDN Virtualization: A Machine Learning Approach

Yeonho Yoo, Gyeongsik Yang, Changyong Shin, Jeunghwan Lee, Chuck Yoo

Korea University, Seoul, Korea

THE 23RD IEEE/ACM INTERNATIONAL SYMPOSIUM ON Cluster, Cloud and Internet Computing

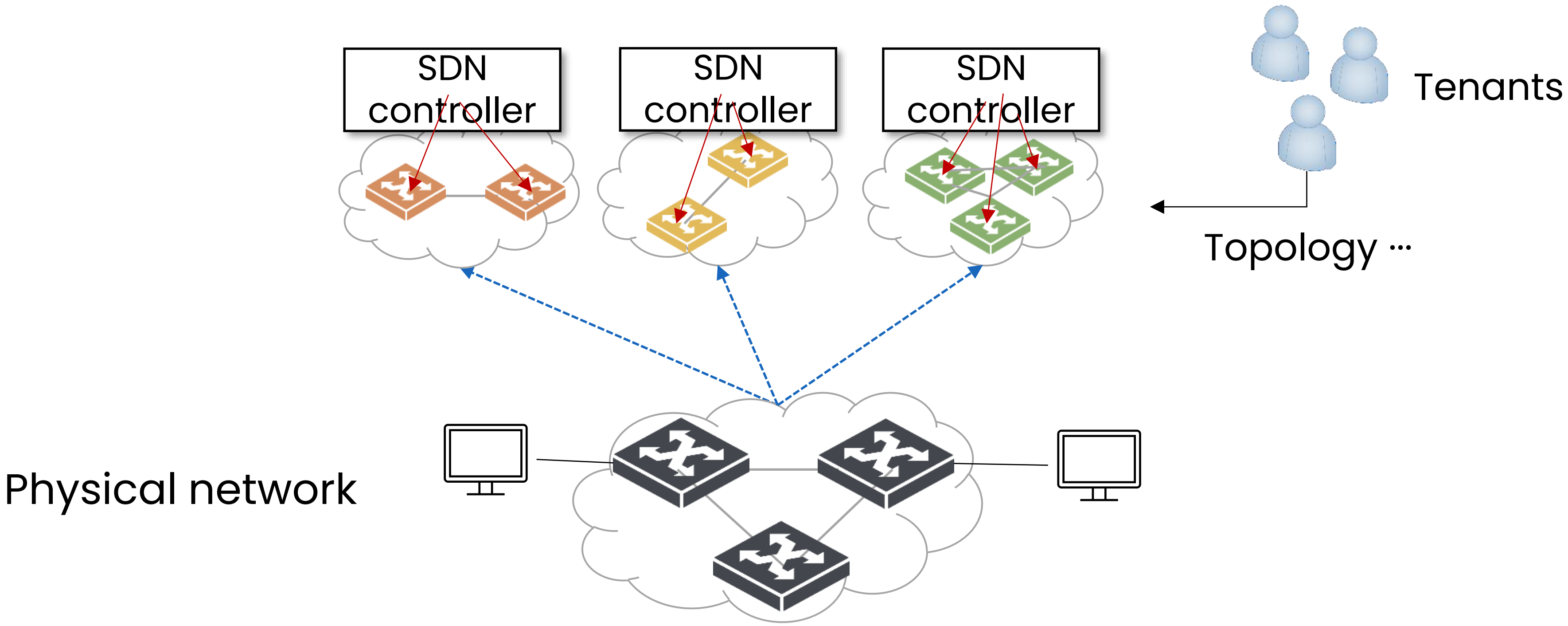
Network Isolation in Cloud

Cloud Infrastructures

- Integral infrastructure for IT services
  - Numerous services on a public cloud
  - Each service is considered a tenant
  - Each tenant desires own computing resources and network
- Network isolation
  - Provide isolated virtual networks (VNs) to tenants
  - Network virtualization (NV) - essential for realizing isolated VNs
- **Software-defined networking (SDN) based NV (SDN-NV)**
  - One promising approach

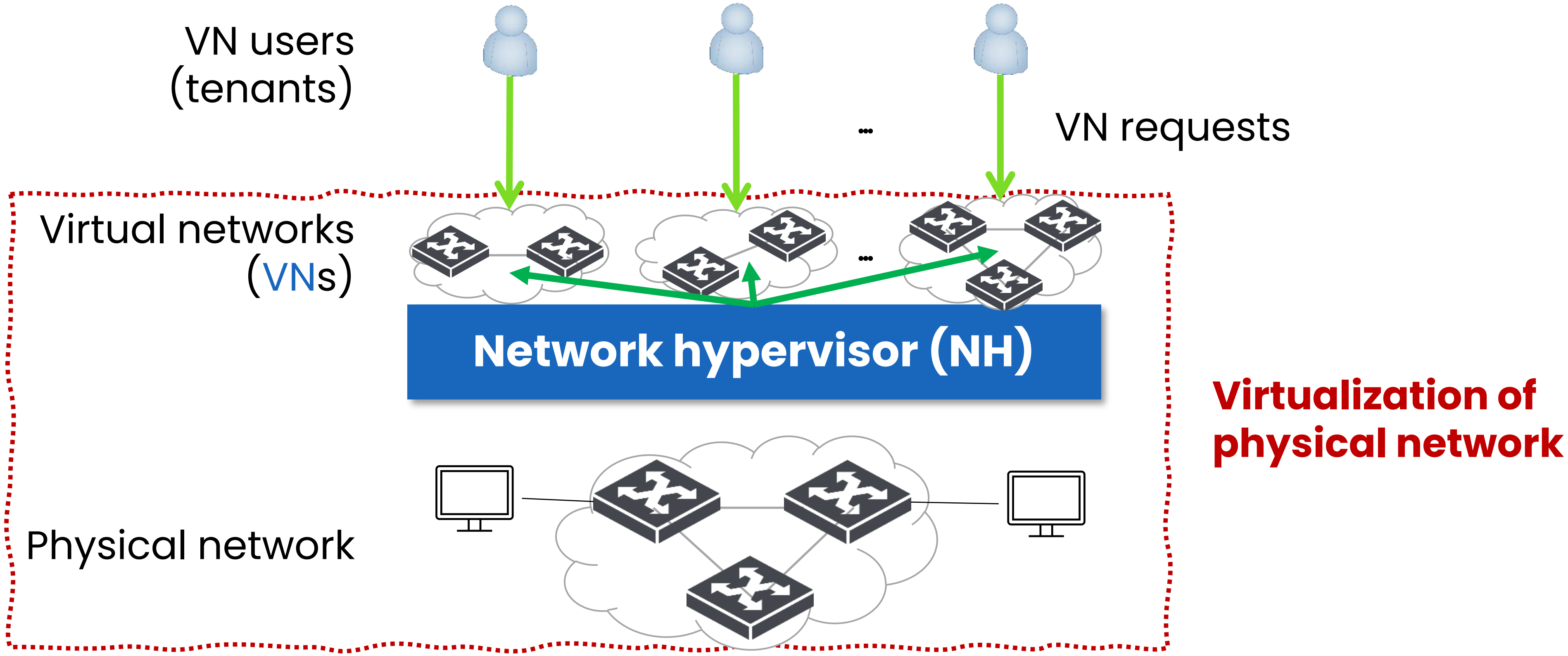
SDN-based Network Virtualization (SDN-NV)

- SDN-NV: support **programmable virtual networks**
  - Tenants create their VN **flexibly** (e.g., topology)
  - Tenants manages their VN **freely** with SDN controllers



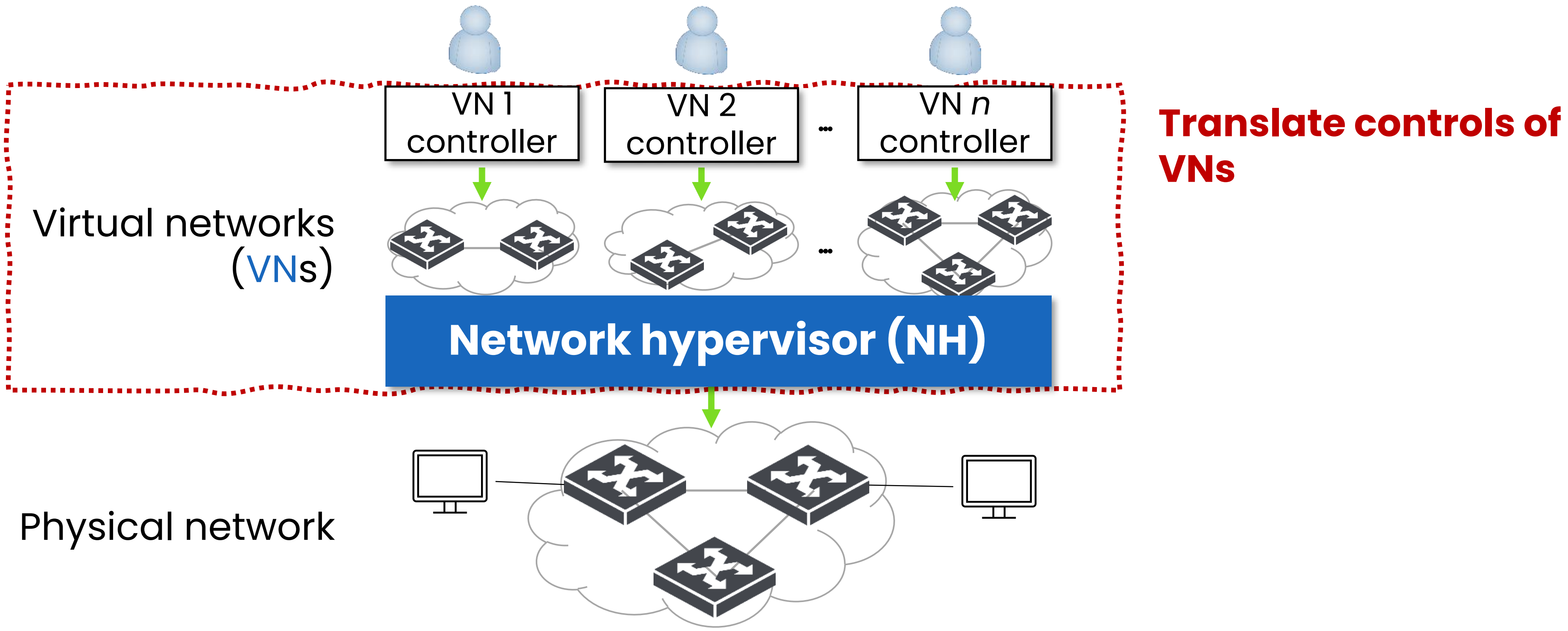
Network Hypervisor (NH): Key Enabler

- Create (virtualize) VNs from physical network
  - Virtual topology with virtual switches (vSwtiches), virtual links (vLinks)



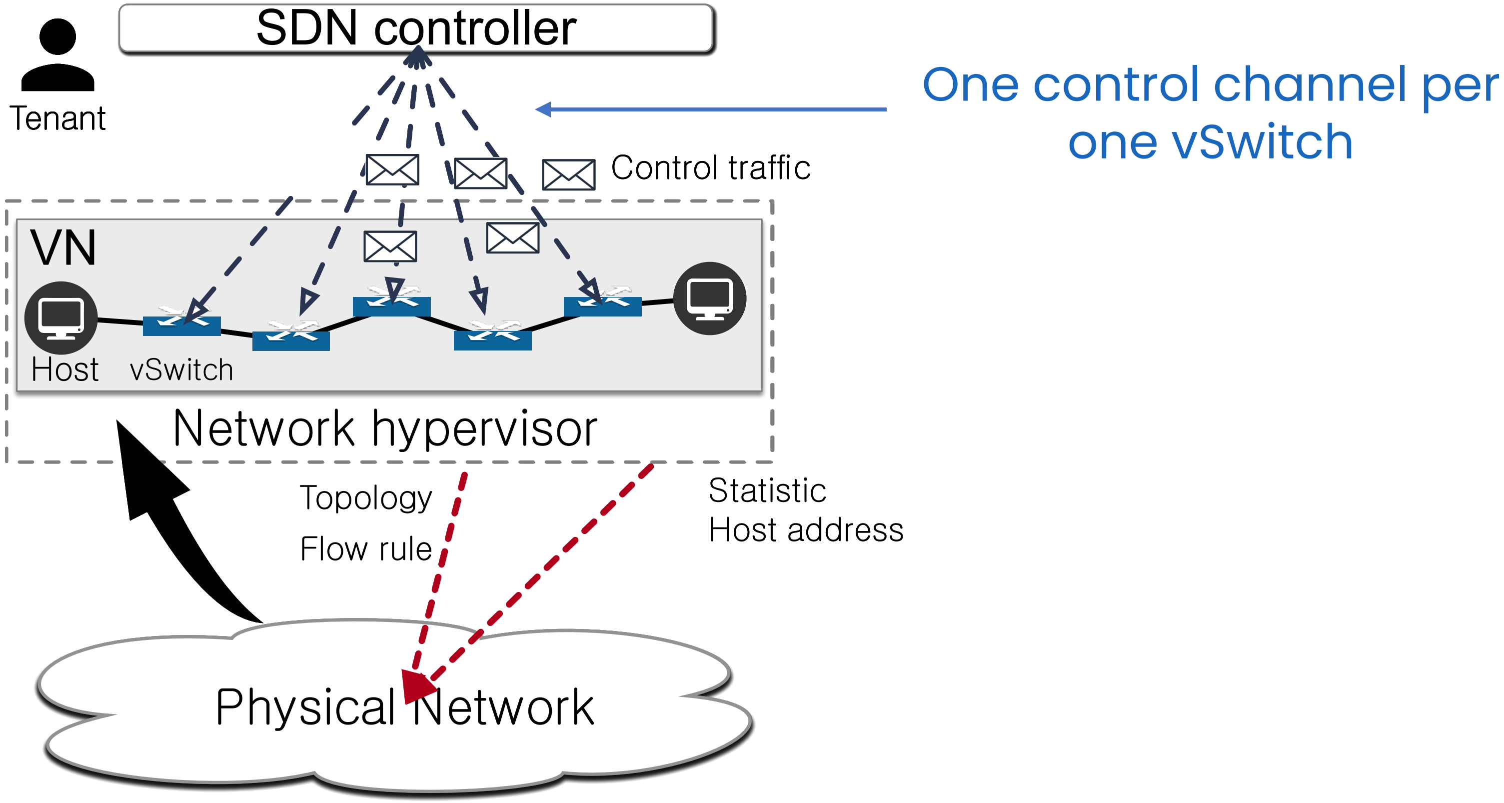
Network Hypervisor (NH): Key Enabler

- Create (virtualize) VNs from physical network
- Translate "control" from SDN controller to physical network



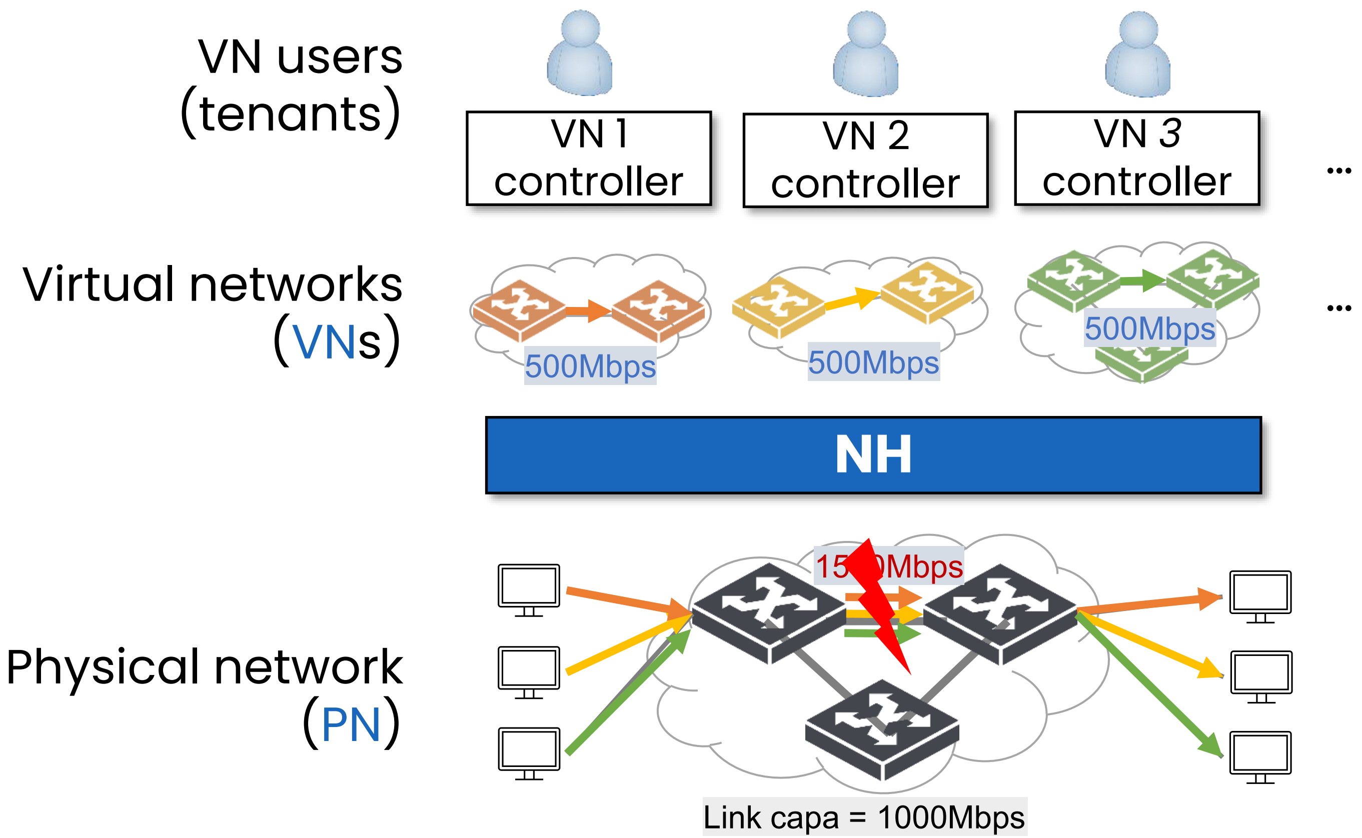
Control Channel for Network Control

- SDN controller sends control messages to NH via **control channel**
  - Network control: Forwarding setup, monitoring, Host address, Topology
- NH **translates** **(virtualize)** various control messages for physical network



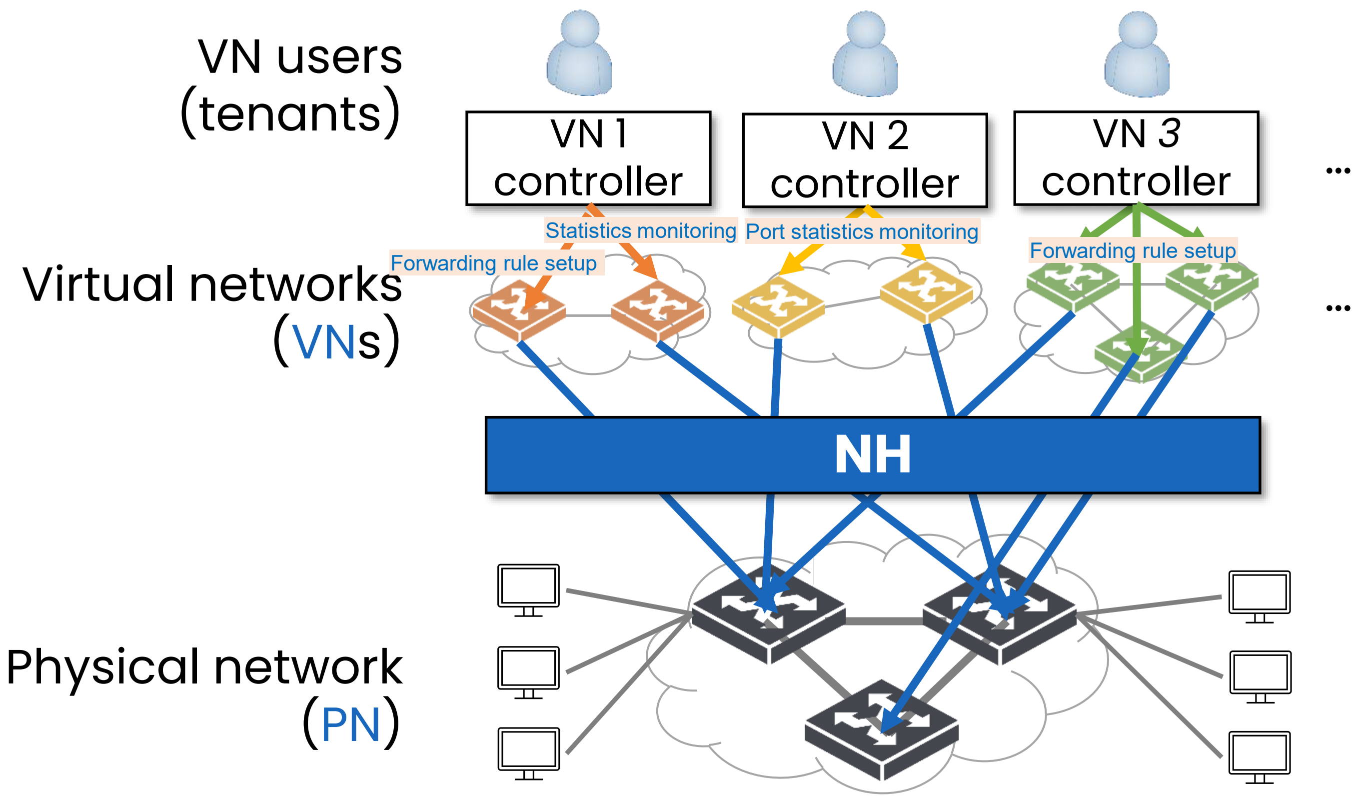
Performance Isolation in Two Planes

- Data plane: bandwidth (link capacity) isolation
  → cannot achieve bandwidth requirements for tenants



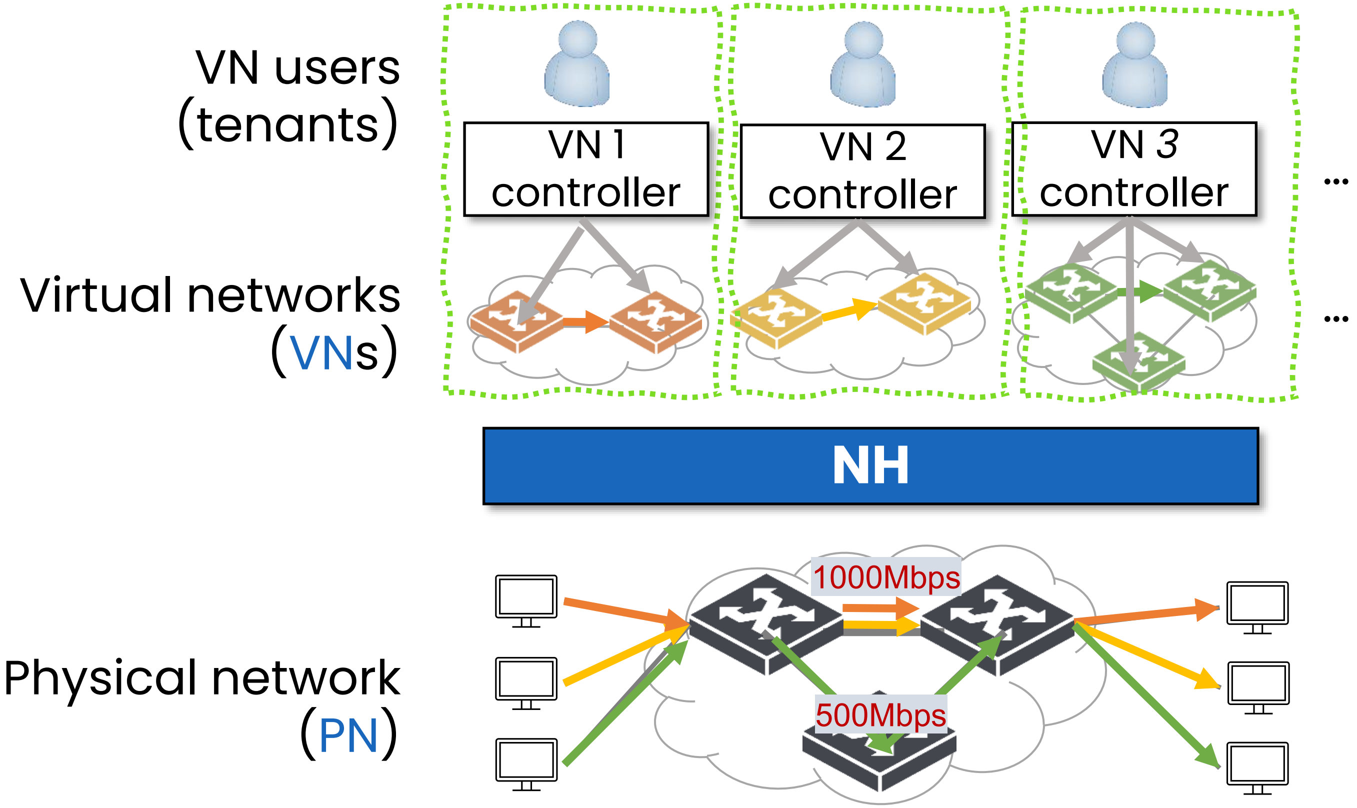
Performance Isolation in Two Planes

- Control plane: control channel isolation
- # of vSwitches, tenants increases
  → Cannot deliver control messages of SDN controllers timely!



Control Channel Isolation: Unsolved Issue

- Two-planes of performance isolation guarantee



Control plane

Not solved yet!

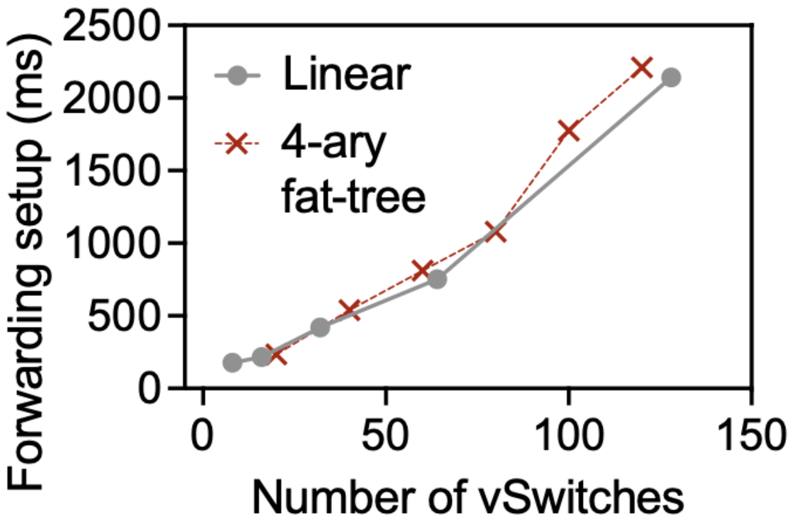
Data plane

TeaVisor
(INFOCOM 2021)

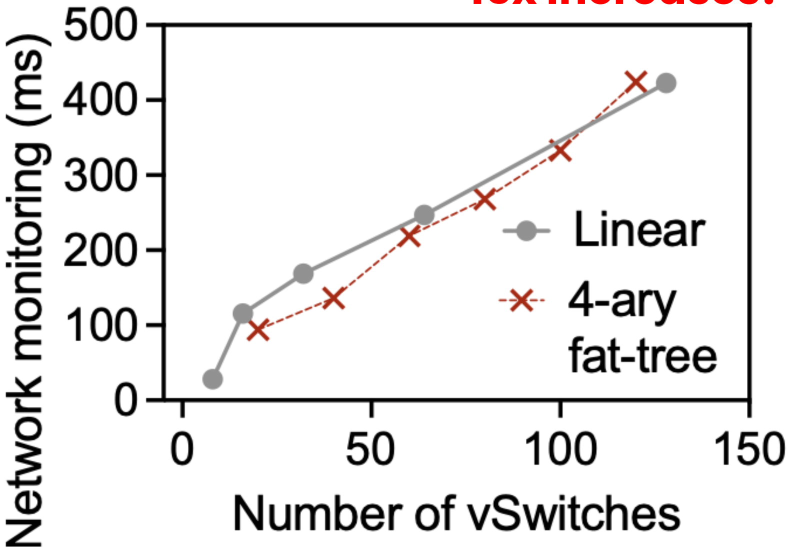
Lack of Control Channel Isolation

- Multiple tenants, VNs, and virtual switches (vSwitches)
  - # of Control channel largely increases
- Network control performance (e.g., latency) degrades up to **15x**!
  - as # of vSwitches increases from 8 to 128

12x increases!



15x increases!



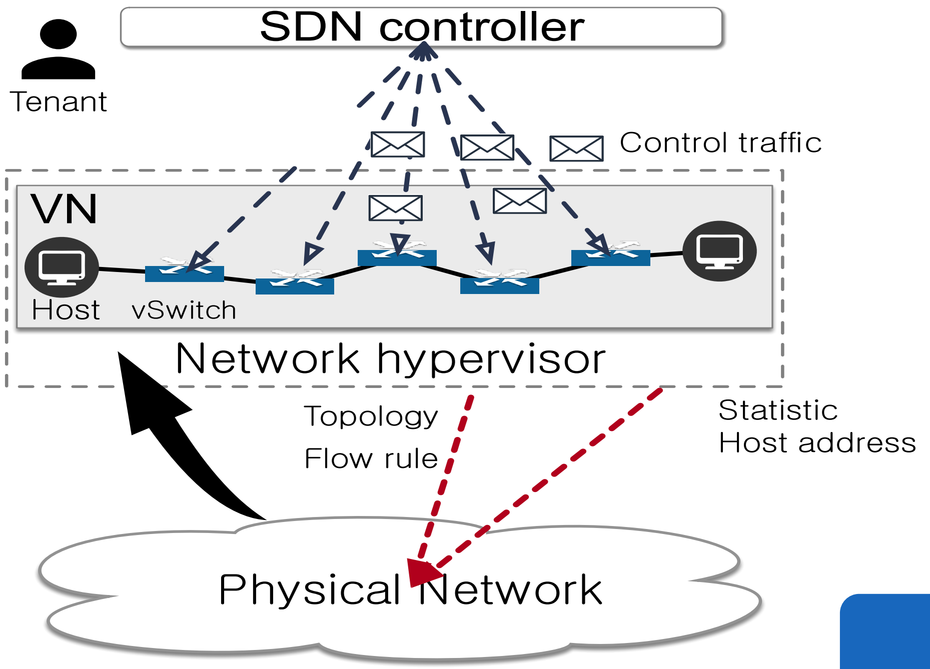
Control channel isolation severely broken!

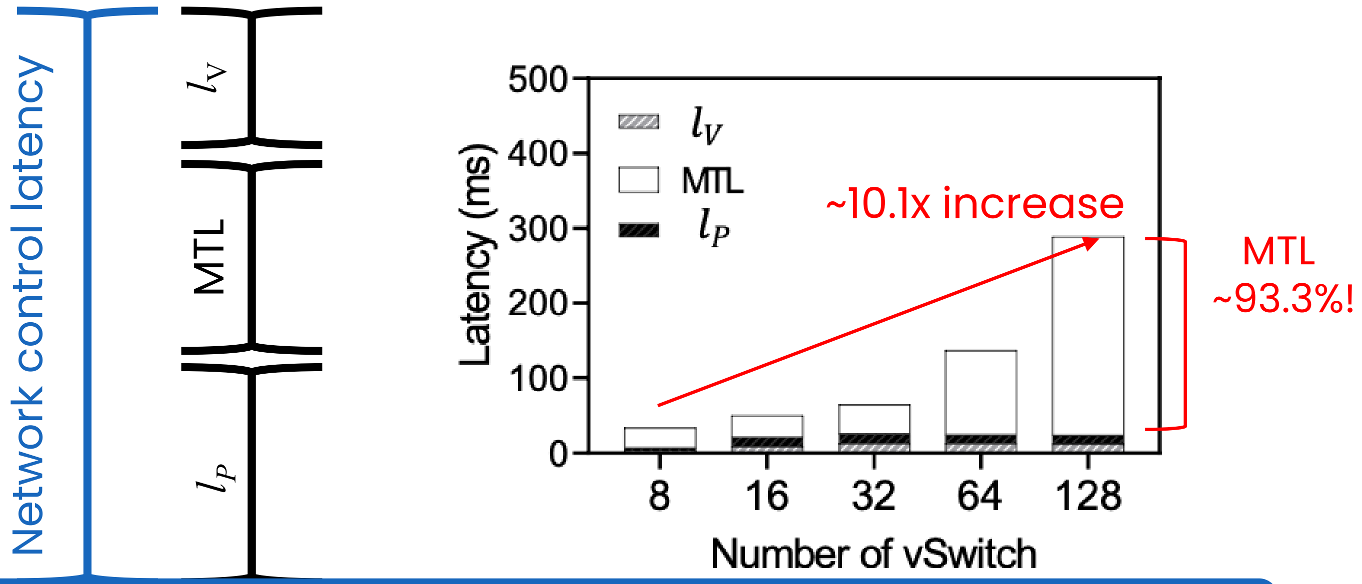
Related Works

	FlowVisor (OSDI 2010)	OpenVirteX (HotCloud 2014)	CoVisor (NSDI 2015)	Libera (IEEE Commag 2020)	V-Sight (IEEE INFOCOM 2020)	TeaVisor (IEEE INFOCOM 2021)
Multi-tenancy	O	O	X	O	O	O
Architecture	Central	Central	Central	Central	Central	Central
Scalability	X	X	X	O	O	O
VM migration support	X	X	X	O	X	X
Network monitoring	X	X	X	X	O	O
Bandwidth isolation	X	X	X	X	X	O
Control channel isolation	**X**	**X**	**X**	**X**	**X**	**X**

Problem Breakdown

- Breakdown of network control latency in SDN-NV
  - L_V: communication latency from SDN controller to virtual switch
  - MTL: control message translation latency within NH
  - L_P: communication latency from NH to physical switch

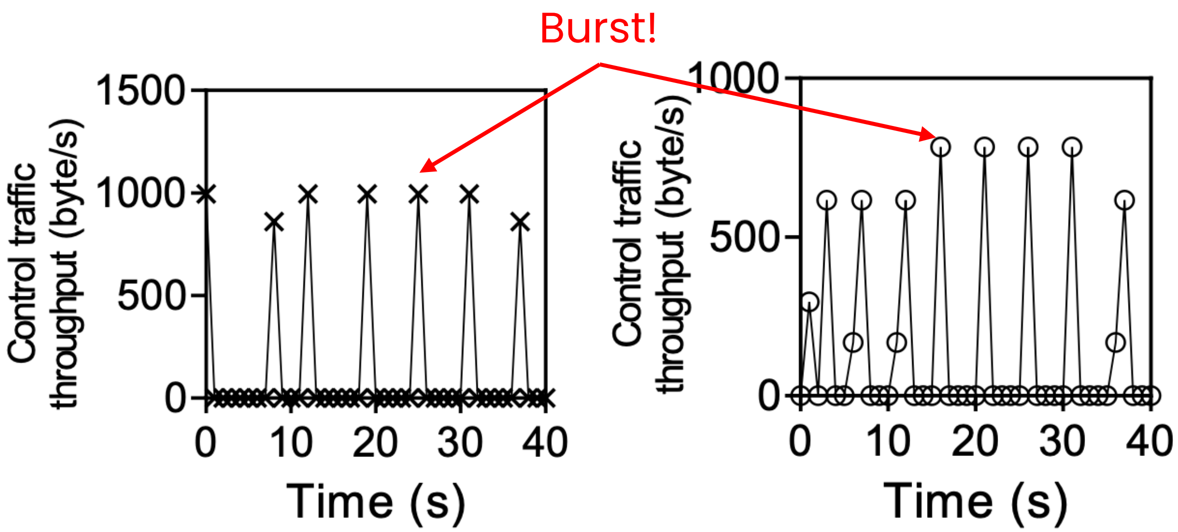


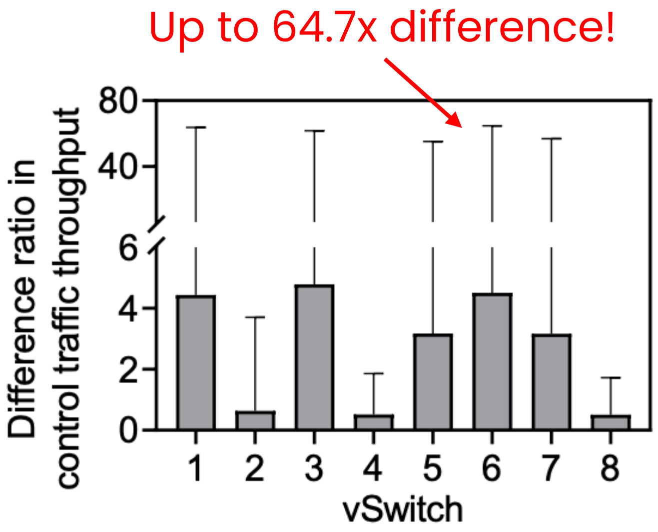


Message translation between vSwitches – the main bottleneck

Cause of Message Translation Bottleneck

- Network control for vSwitch: burstly generated
  - ✓ Burst traffic generated according to specific events and messages
  - ✓ Hard to handle large network controls at the same time
- The amount of control traffic varies greatly by switch
  - ✓ vSwitches showing ~64.7x difference in network control amount
- Our approach: **Per-vSwitch isolation of network control**





Two Approaches in vSwitch Control

Reactive control (feedback-based)

- Allow the translation power of NH according to incoming network controls
- Difficult to rectify once burst control traffic has occurred
- Too late to react when already interference exists

Proactive control

- Control the translation power of NH in advance
- Possible to prevent bottleneck (burst traffic) in advance
- **Accurate prediction** required

Machine Learning for Proactive Control

- Reactive control vs **Proactive control**
  - Adjust control traffic translation amount for NH in advance

- How to predict control traffic?
  - Machine learning (ML) appraoch
    - ✓ML model can predict control traffic as time-series sequence
    - ✓ML model can predict control traffic per vSwitch
    - ✓ML model can adopt at various settings

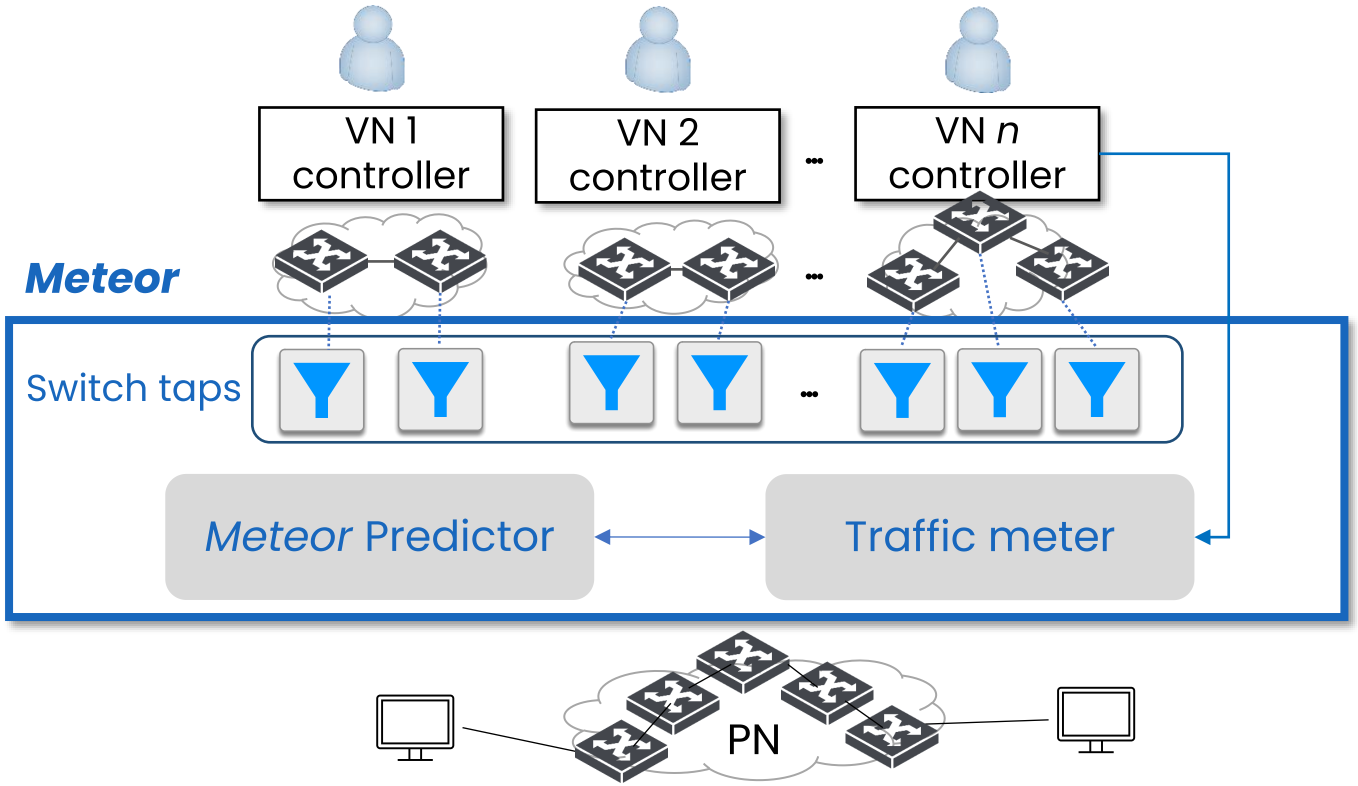
Meteor: New NH for Control Isolation

- Control channel isolation per vSwitch by prediction

✓ **Predict** control traffic of vSwitch in advance

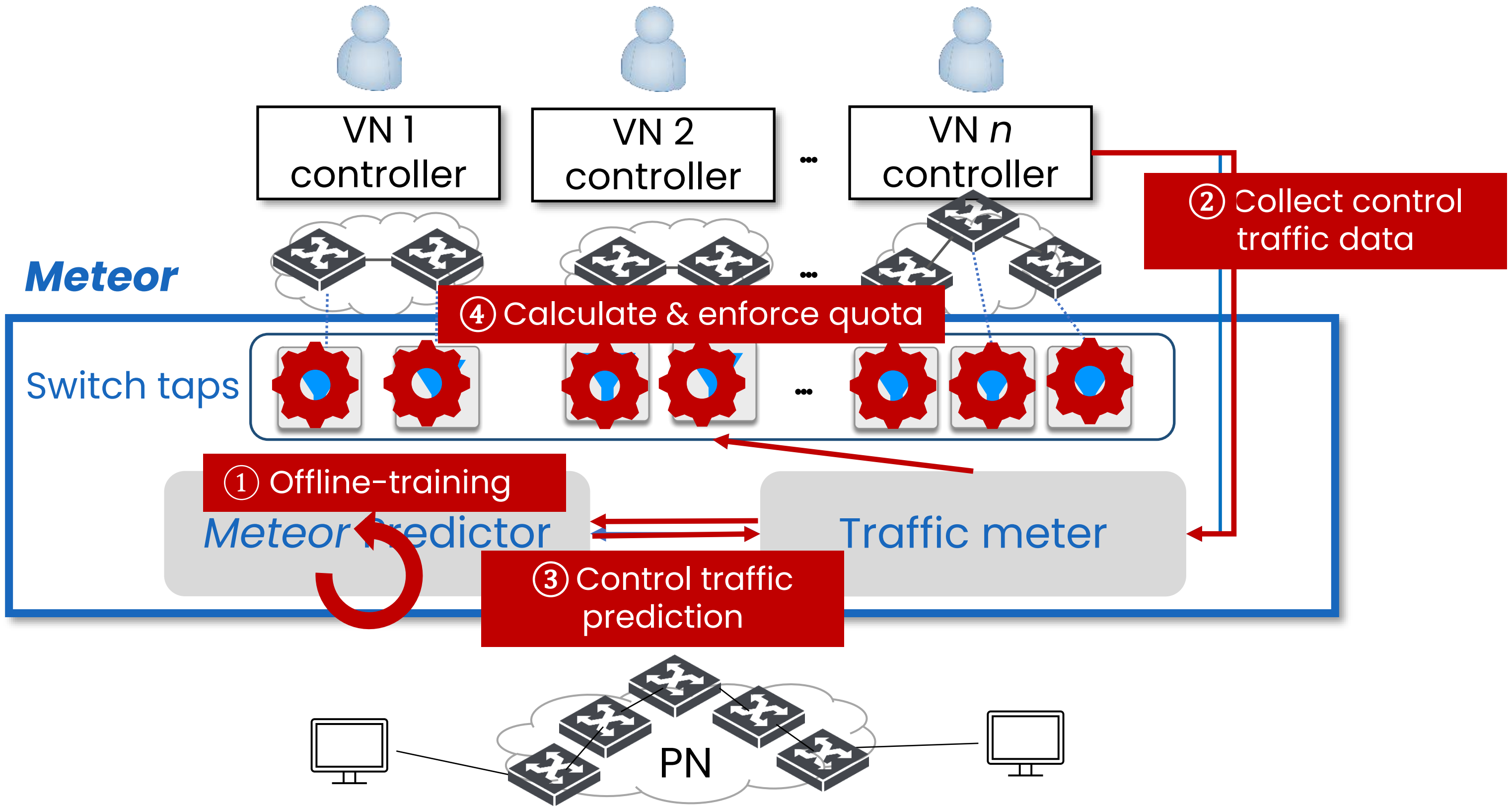
✓ Derive the proper translation quota (**quota**)

✓ **Enforce** the quota



Meteor Workflow

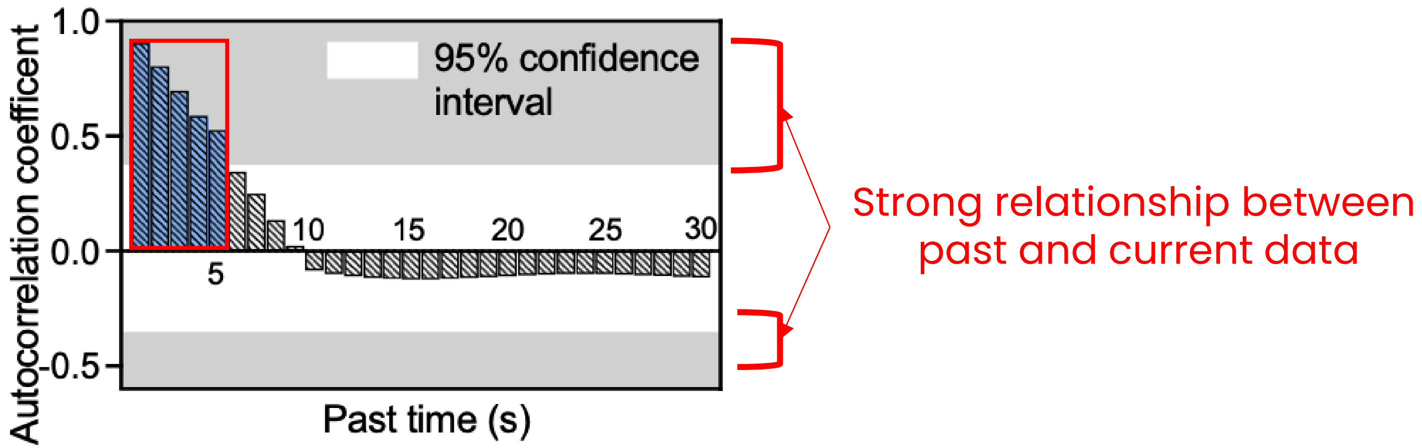
Meteor is new NH and composed of
Meteor predictor, Traffic meter, Switch taps



Meteor Predictor

- Q: Which ML model is suitable?

  ✓ Predict time-series of the control traffic
  ✓ Correlation (autocorrelation) between past and current control traffic

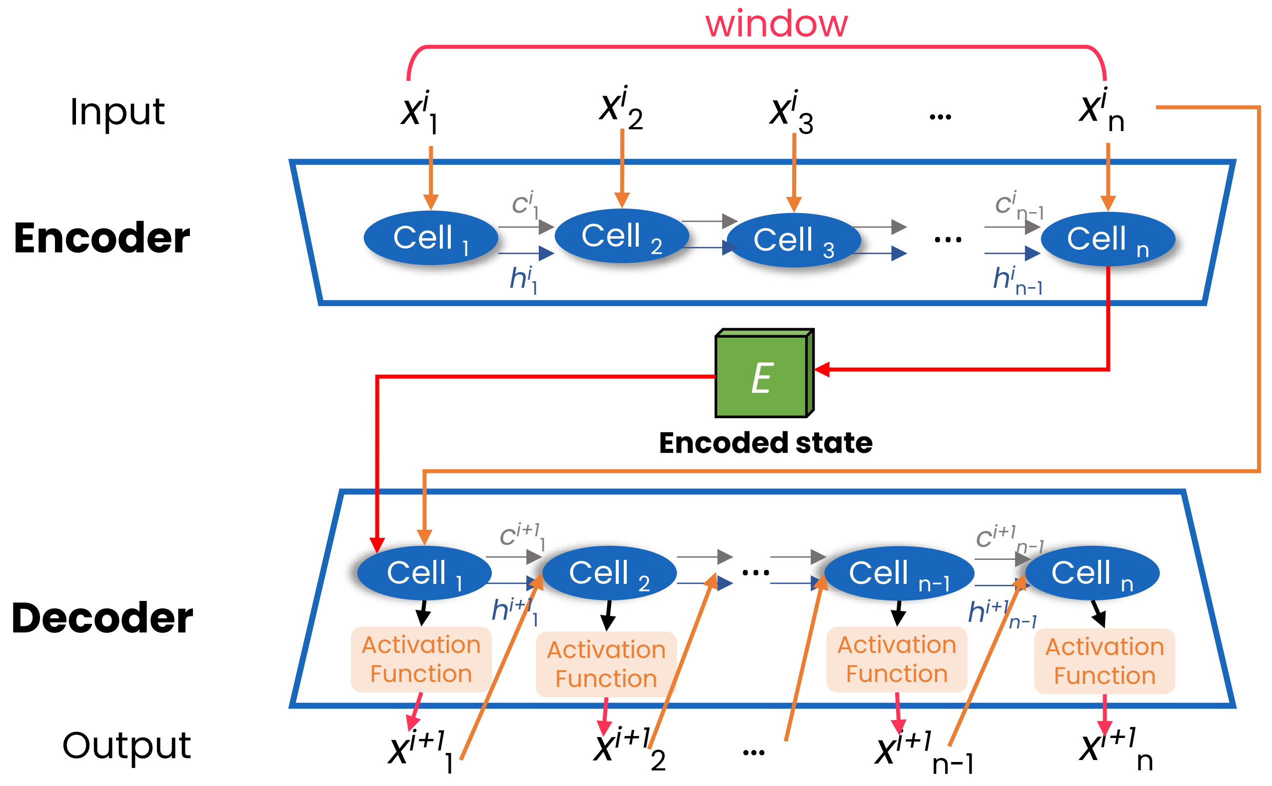


➢ Control traffic of the past 5s has strong correlation with the current
➢ Short time-series prediction (generation) model → Autoencoder
➢ LSTM-Autoencoder - the best in terms of accuracy (see evaluation)

* Please check our paper for detail

Meteor Predictor as LSTM Autoencoder

- Predict future control traffic by past control traffic
  - LSTM cell: cell state (c) and hidden state (h) to keep "time-series"
  - Encoder: generate an **encoded state** that compresses the input data
  - Decoder: predict the output for the encoded state



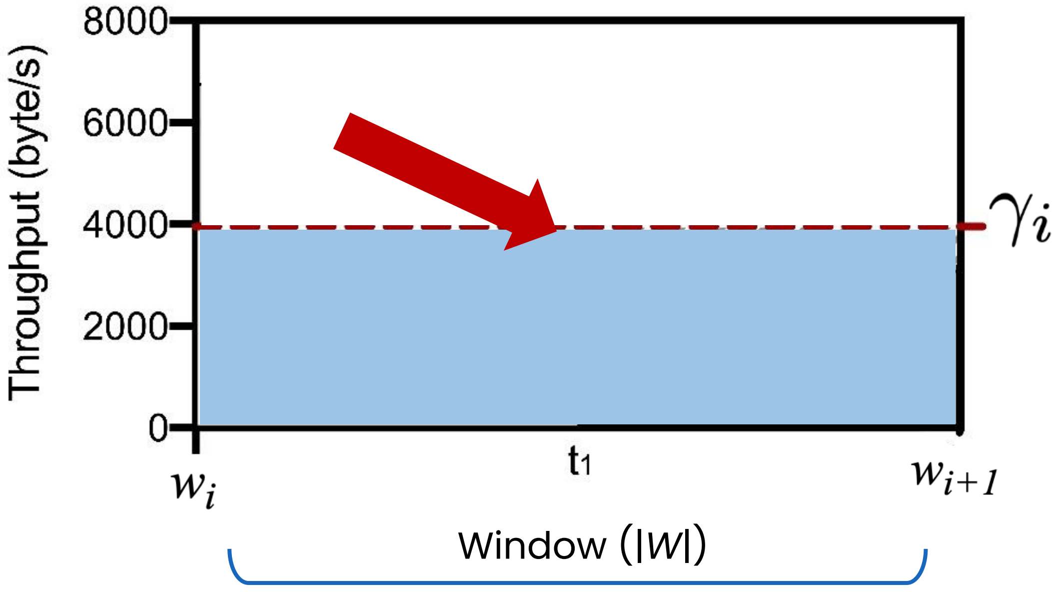
* Prediction window set as 2 s (empirically)

Meteor Predictor Details

- Input features for prediction
  - Amount of network control messages per type
    - E.g., new traffic arrival, flow rule installation, network statistics
  - Features for network topology
    - E.g., # of hosts, # of active links
- Hyperparameters
  - Carefully optimizes 7 hyperparameters by random search with wide range
- Training dataset
  - Self-generated for predictor training
  - Can be generated for various SDN controllers
- More details in our paper

Quota Derivation

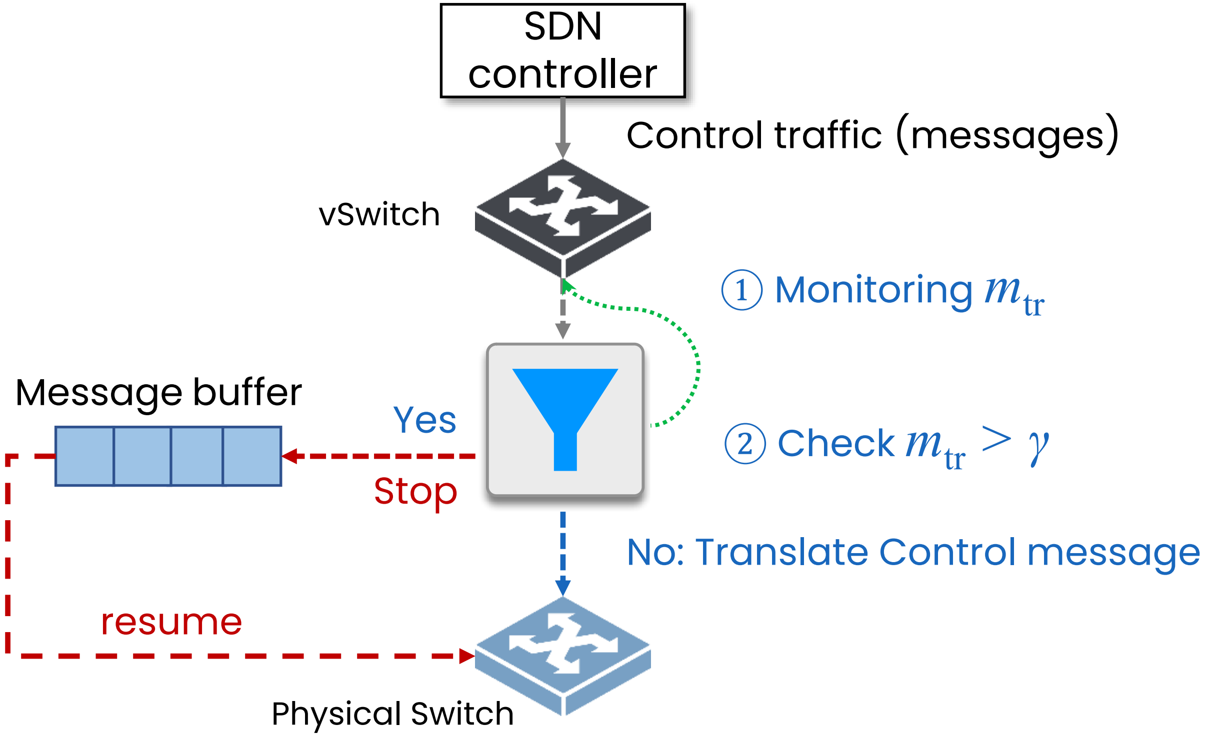
- Derive the "proper" quota (γ) for message translation per vSwitch
- Find γ for processing the predicted amount during the window uniformly
- Calculate γ when the areas of **H and L** are equal
  - Area: the amount of network control predicted



$$\int_{w_i}^{w_{i+1}} P_i(t)\,dt = |W| \times \gamma_i$$

Quota Enforcement

- Enforce γ for translating control messages (by switch tap)
  - Monitor control channel throughput (m_{tr})
  - $m_{\mathrm{tr}} < \gamma$ → translate immediately
  - $m_{\mathrm{tr}} > \gamma$ → buffer the messages and translate later



Evaluation

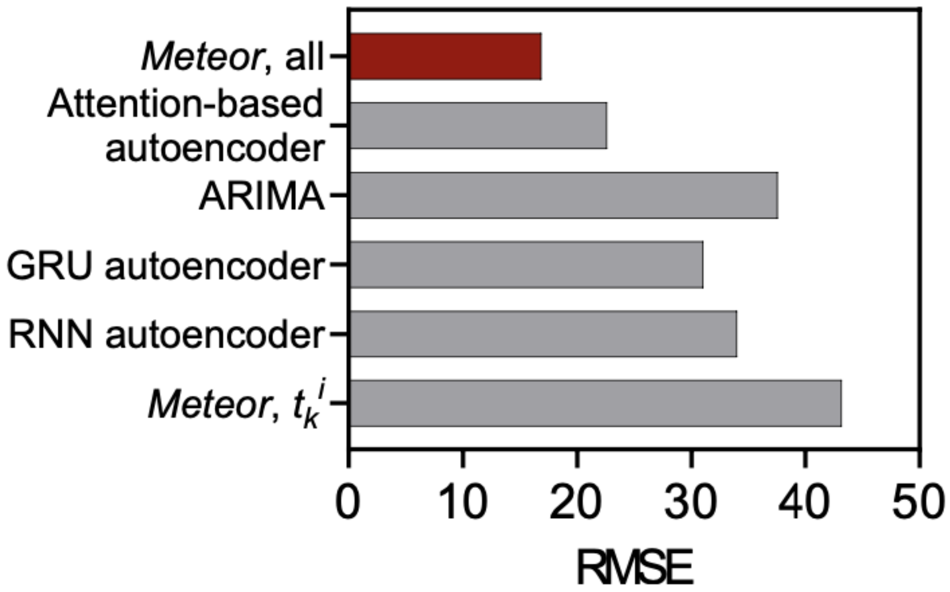
- Setting
  - Physical network, NH (*Meteor*), SDN controller at individual server
    - Intel Xeon E5-2600 CPUs and 64 GB memory
  - *Meteor* predictor trained by NVIDIA RTX 2080 Ti GPU

- Comparison
  - Libera: Latest open-source NH
  - Libera-RC: Libera with reactive control
    - Adjust quota without prediction (only based on past data)
  - *Meteor*

- Metric

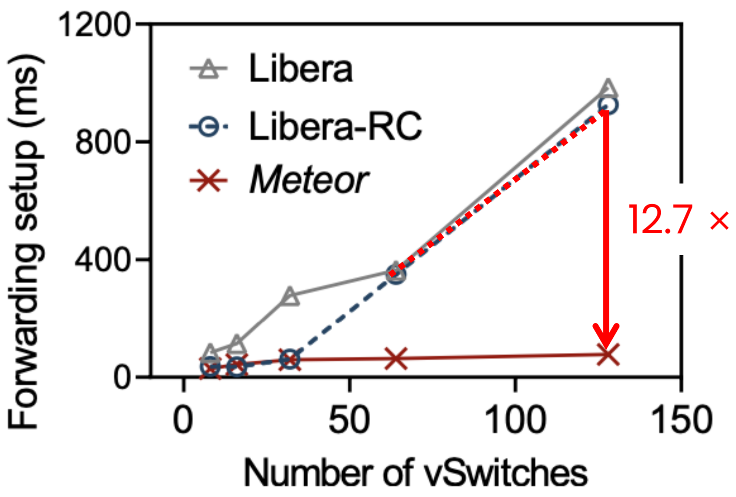
  E1: Prediction accuracy
  E2: Control channel isolation
  E3: Generalizability
  E4: Overhead

E1: Prediction Accuracy

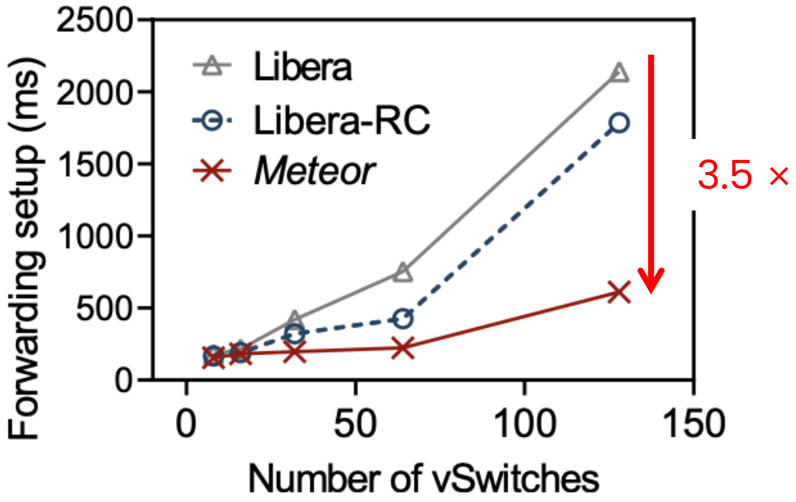


- Compare prediction accuracy (RMSE) between various model structures
- *Meteor* predictor outperforms other ML models: ~ 2.6x better

E2: Control Channel Isolation



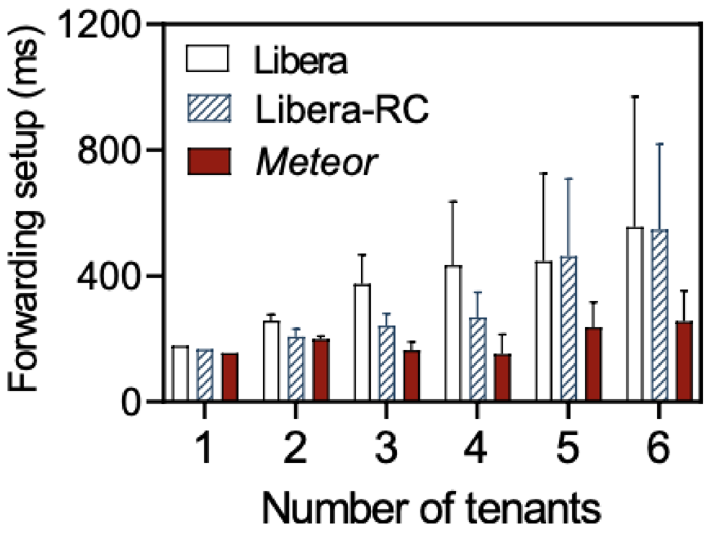
Message translation latency in NH



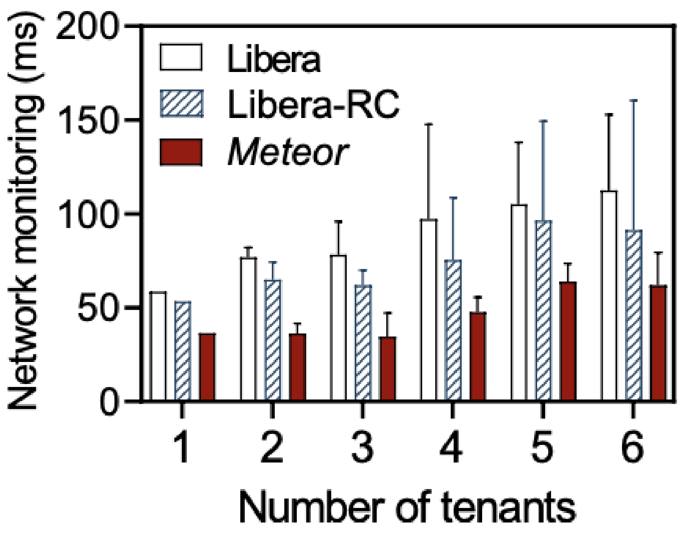
Network control latency
(controller-NH-physical switch)

- Message translation latency: ~12.7x better
- Network control latency: ~3.5x better
- Reactive control is not enough to large-scale networks (over 50 vSwitches network)

E3: Generalizability



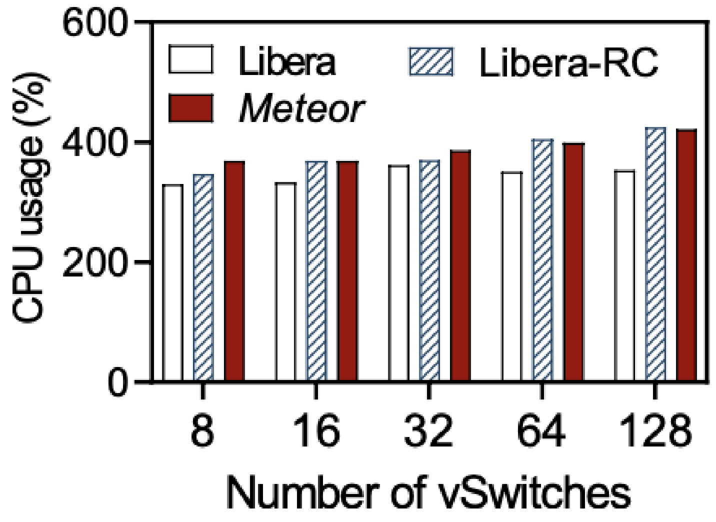
Forwarding setup latency



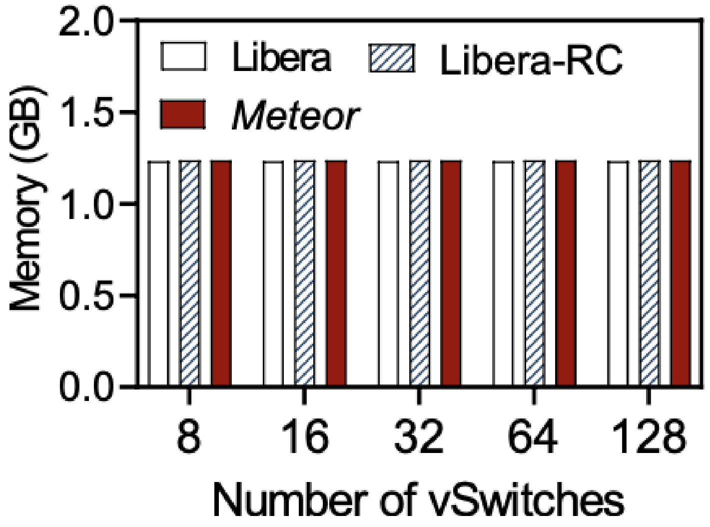
Networking monitoring latency

- *Meteor* predictor trained by a restricted setting
- Apply on different and bigger settings (4-ary fat tree topology & multi-tenants)
- Meteor: ~2.2x, ~3.8x better for forwarding setup and network monitoring latencies

E4: Overhead



CPU usage



Memory usage

- Consume ~12% more CPU cycles
  - For vSwitch monitoring and quota enforcement
- Consume 0.4% (5.3 MB) more memory
- Considering performance improvement, the overhead is reasonable

Conclusion

- **Tackle control channel isolation in SDN-NV**
  - Essential property of isolated VNs for tenants
- **Contributions**
  - Predictive approach for isolation
  - Improves message translation and network control latencies ~12.7x and 3.8x
  - Applicable to complex network settings with reasonable overheads

- **Try *Meteor* !!**

Open Research Objects (ORO)

Research Objects Reviewed (ROR)

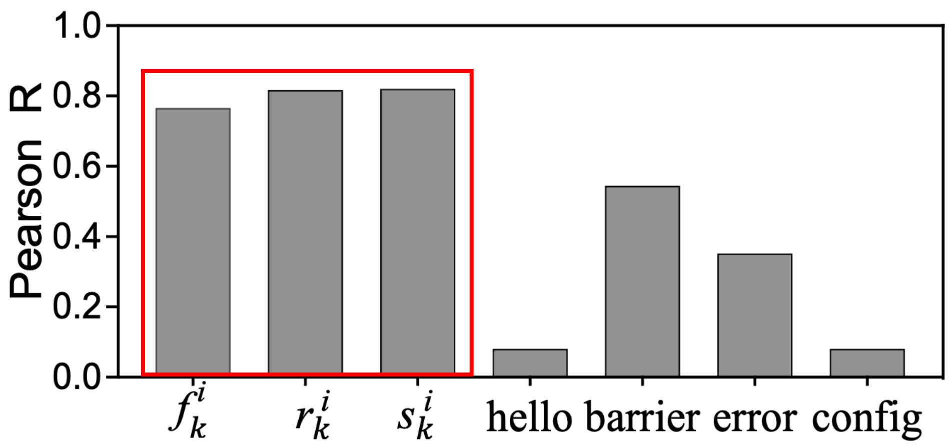
github repository

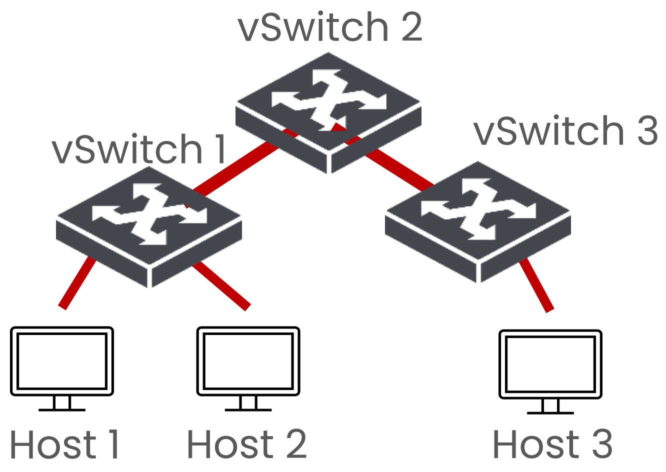
Thank you for listening

Yeonho Yoo
yhyoo@os.korea.ac.kr

Input features

- Input features (x)—not only control traffic!
  1. Control messages most relevant to control traffic by Pearson R
     - We choose Control traffic (t), new traffic arrival messages (f), flow rule installation messages (r), network statistics messages (s)
  2. Features that represent data plane of vSwitch
     - # of hosts (d), # of active links (a)
- Traffic meter collects each vSwitch`s input features ($x = \{t,f,r,s,d,a\}$) per 100ms

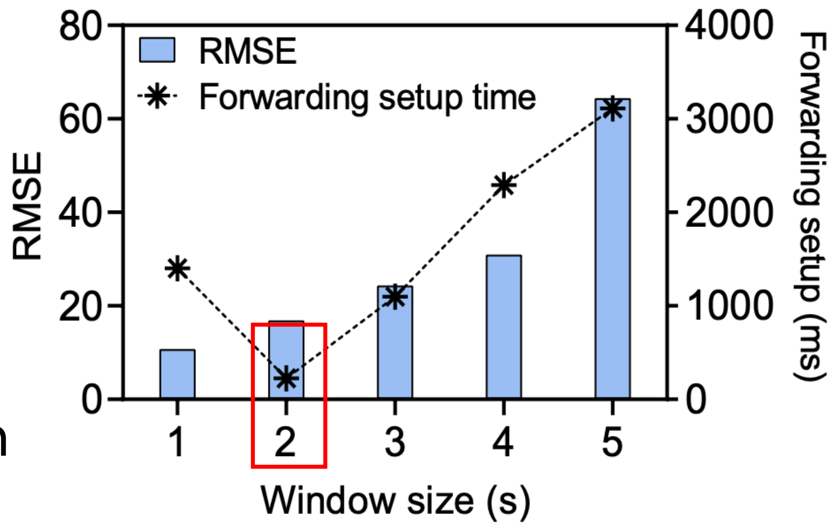




	vS1	vS2	vS3
d	3	2	2
a	2	0	1

Model Parameters and dataset

- Window size
  - Window: unit of time to predict at once
  - Carefully select 2s by our experiment
- Hyperparameters
  - Carefully select 7 parameters by random search
- Training dataset
  - Generated by ONOS controller and linear VN topology
  - Vary # of vSwitch from 1 to 128
  - Data record interval: 100ms
  - 360K dataset records



Hyperparameter	Search range	Best value
# of Encoder / Decoder layer	1-3	2
Hidden State size	16,32,64,128	64
Activate function	Linear, ReLU, leaky ReLU	Linear
Optimizer	ADAM, SGD	ADAM
Batch Size	120, 240, 360, 720, 840	720
Learning rate	0.001, 0.003, 0.00, 0.03	0.01
Epoch	100, 200, 400, 1000	200