A Demand-aware Networked System Using Telemetry and ML with ReactNET

Seyed Milad Miri, Stefan Schmid, Habib Mostafaei

TECHNISCHE UNIVERSITÄT BERLIN

TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Presented by Max Franke

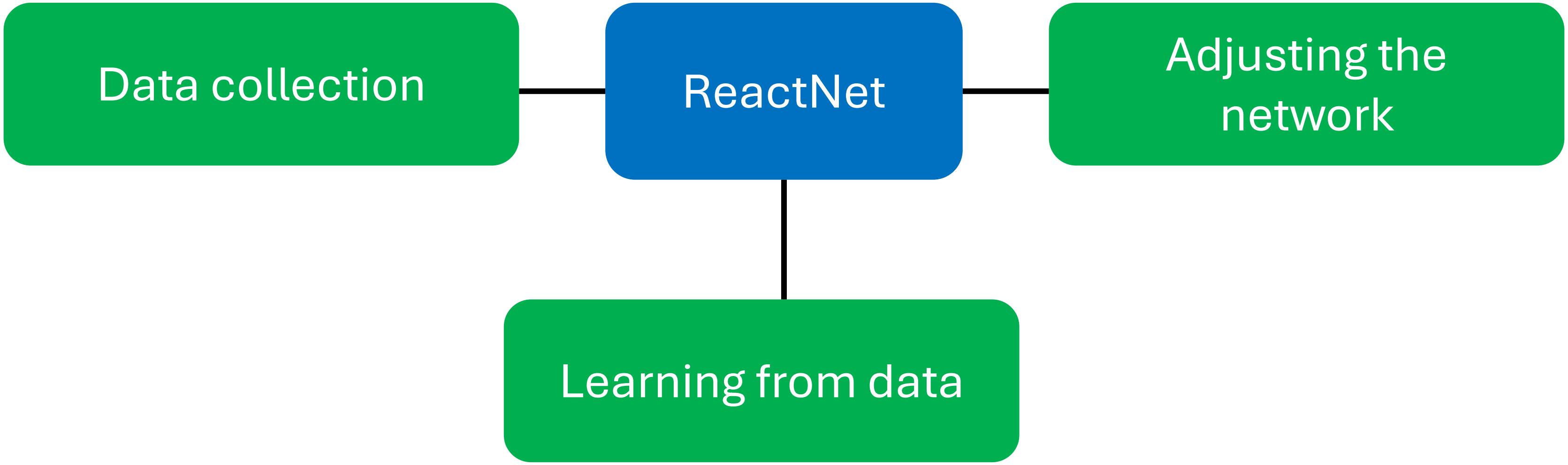
Challenges

- Automating network management operations is challenging and error-prone
- Meeting the QoS/QoE requirements of the applications, e.g.,
  - latency-critical applications like online gaming
  - bandwidth-hungry applications such as distributed AI

Motivation

- Use of programmable networks to collect fine-grained telemetry information
  - no extra server for data collection
  - enable per-packet telemetry collection
- Leveraging ML techniques to learn from the flows and adjust the network configurations

ReactNet components



Data collection

- Leverage programmable networks for telemetry collection
  - rich set of telemetry information
  - accurate
  - line-rate data collection

Feature	Size in bits
Ingress port	9
Flow_interval_time	48
enq-qdepth	19
deq-qdepth	19
deq-timedelta	32
Protocol number	8
Source port	19
Destination port	19
IPv4 source address	32
IPv4 destination address	32

ReactNet collects data from the packets of flows when needed

- reduces the overhead
- operator can enable this using a flag

Data collection

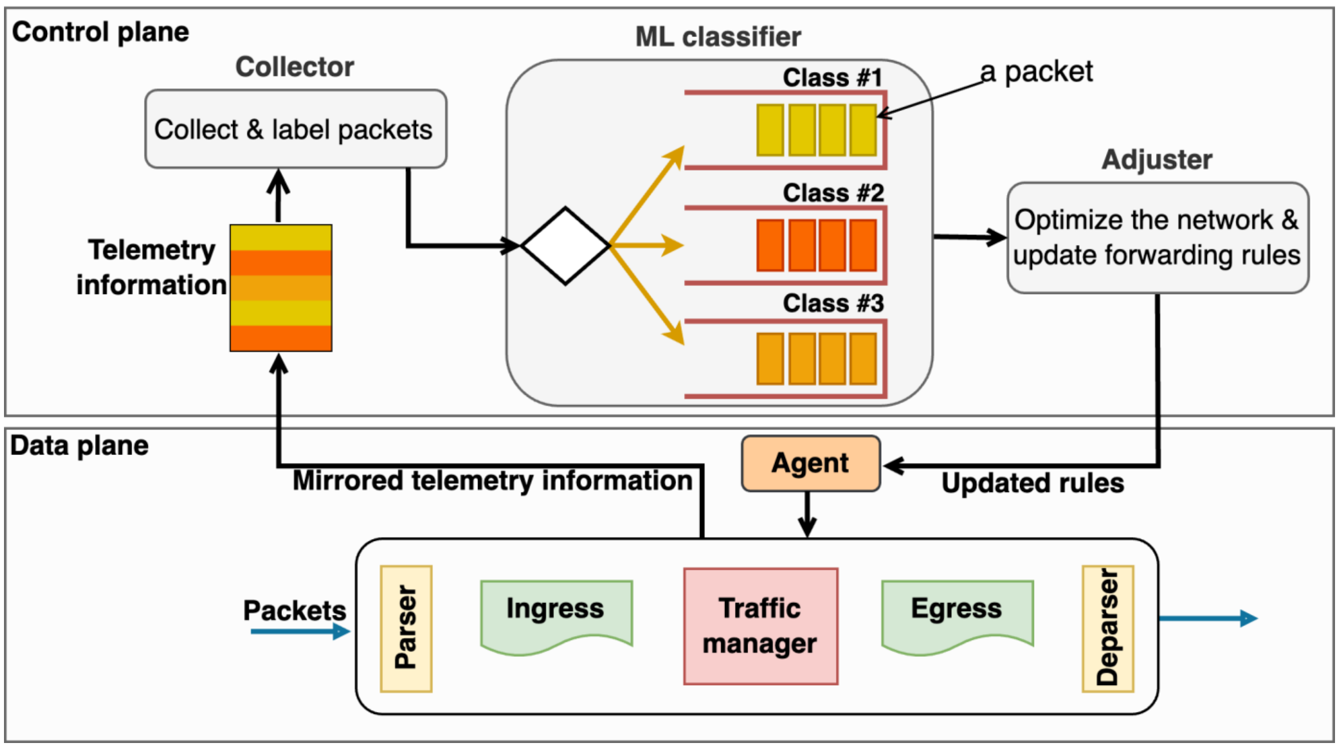
Learning from data

- Classify packets of different flows
  - label packets based on the desire of network operators to classify them
  - learn from data while capturing them

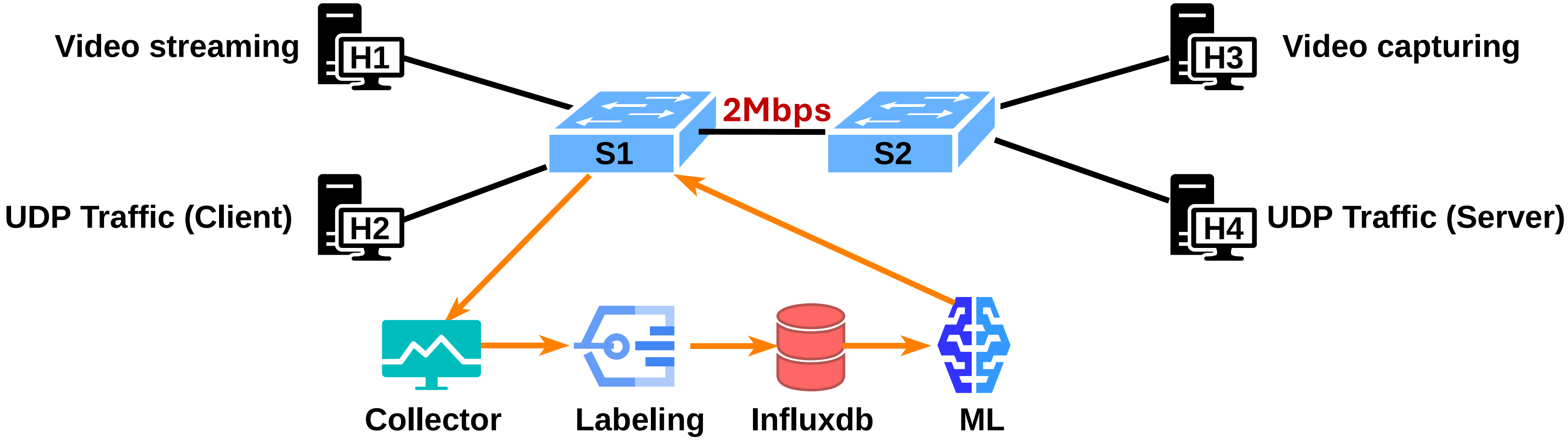
Adjusting the network

- Adjust the system into new state
  - assigned the desired priority to the packets of each flow
    - through the provided APIs of the programable switches
  - adjust forwarding rules

ReactNet architecture



Testbed network



ReactNet performance evaluation

1. Video streaming scenario
2. Accuracy of ML Classifiers

Video streaming scenario

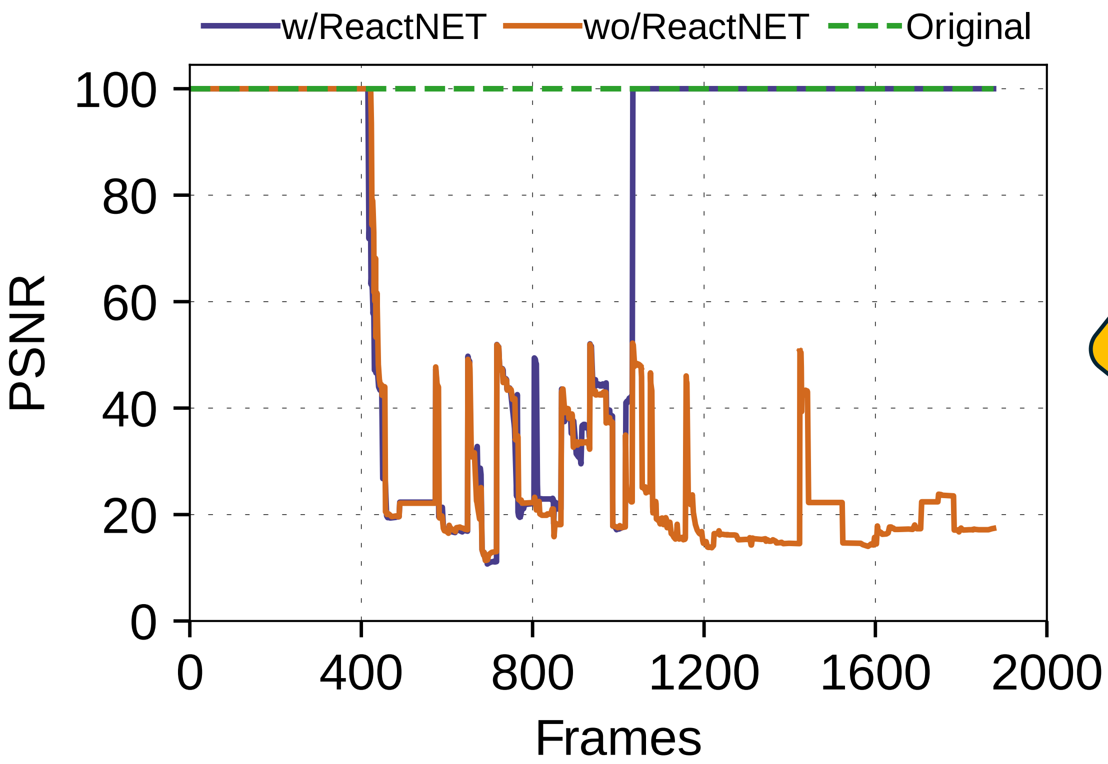
- Impact on the image quality metric:

$$\mathrm{PSNR}(f,g) = 10\log_{10}(255^2/\mathrm{MSE}(f,g))$$

- where

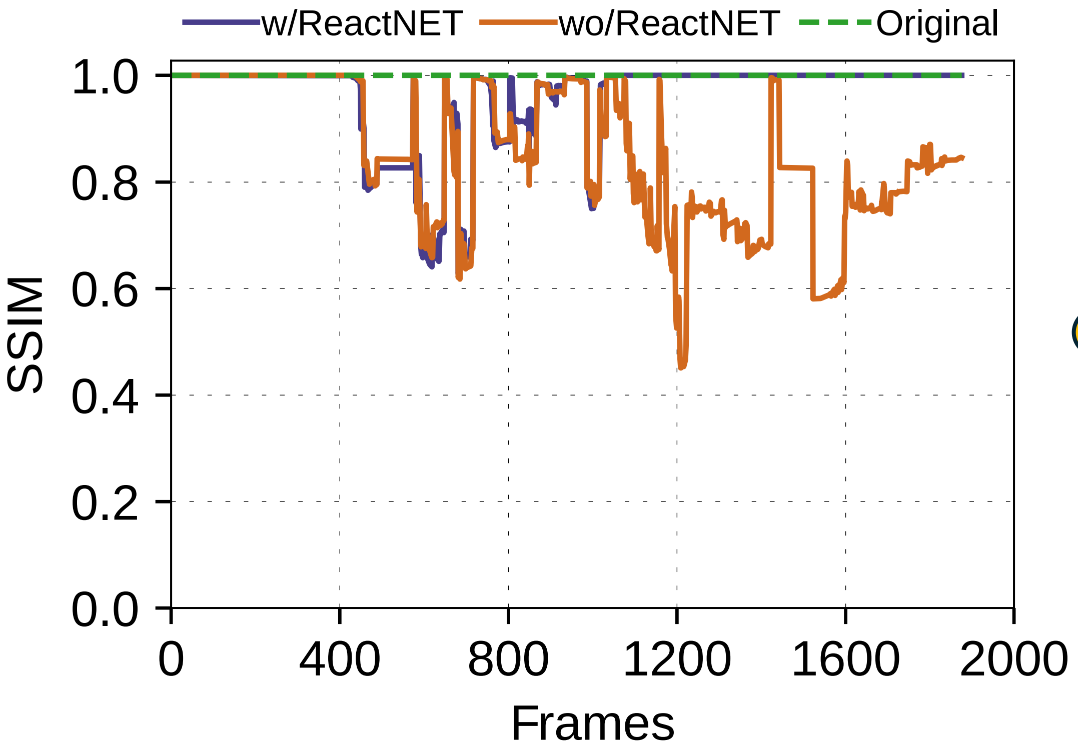
$$\mathrm{MSE}(f,g) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f_{ij} - g_{ij})^2$$

The effect of ReactNet on the PSNR metric



after 10 seconds

The effect of ReactNet on the SSIM metric



after 10 seconds

Impact of ReactNet on video quality

Video output without using ReactNet

Video output after using ReactNet

Accuracy of ML classifier trace

- ~600k packets from IoT devices [10]

Name	Class
Energy	0
Appliances	1
Hubs	2
Health-Monitors	3
Cameras	4
Others	5

Accuracy of ML classifier

Model	Accuracy	MSE	Precision
Decision Tree	0.99	0.1	0.75
K-Nearest Neighbors	0.99	0.99	0.87
Random Forest	0.98	0.99	0.57

Conclusion

- ReactNet is built on two key enabler technologies: programmable networks and machine learning
- Programmable networks can learn from packets and tune the network
- Supervised ML-based classifiers achieve high accuracy when applied on traffic flows