Realizing network challenges for Network Slicing

Pablo Serrano

http://www.it.uc3m.es/pablo/

Instituto de Computação - UFF. Sept. 3rd, 2020

uc3m | Universidad Carlos III de Madrid

About me

Dr. Pablo Serrano (IEEE SM)

Associate Professor, Univ. Carlos III de Madrid (UC3M)

Past visiting positions at:

Univ. Massachusetts Amherst, Univ. of Edinburgh,
Trinity College Dublin, Telefónica R+D Barcelona,
Univ. Roma Tor Vergata (currently)

Current project:

- 5G-EVE: European 5G Validation platform for Extensive trials

Research interests:

- 5G, Wireless Communications, Performance Analysis, Energy Efficiency, Experimental Research, Testbeds

About UC3M

- Universidad Carlos III de Madrid (UC3M)
  - Act of the Spanish Parliament on 5 May 1989
  - First Chancellor was Professor Gregorio Peces-Barba
  - Approx. 20k students
  - Highest average grade achieved by students in Madrid

- Internationalisation
  - 20% of students at UC3M are foreign
  - Higher at both master's (30%) and doctoral (43%) levels.
  - 51% graduates have participated in international mobility programmes
- Among the top 150 best universities for employability
  - It has risen by 20 places in the QS Graduate Employability Ranking 2020
  - 92,3% found work in the first year after graduation.
- Amongst the best universities worldwide in 6 fields (incl. CompSci)

Project involvement

- I have been involved in several H2020 projects that trailblazed a new family of concepts
- 5G-NORMA
  - Network Softwarization at all layers, including RAN (cf. xRAN and ORAN)
- 5G-MoNArch
  - Big Data Driven Networking
  - Application of AI to network management (cf. ETSI ENI)
  - Elastic resource management (cf. Rel 17 study items)
- 5G EVE, 5G VINNI
  - Large scale testbeds

Contact

Pablo Serrano

Univ. Carlos III de Madrid (UC3M)

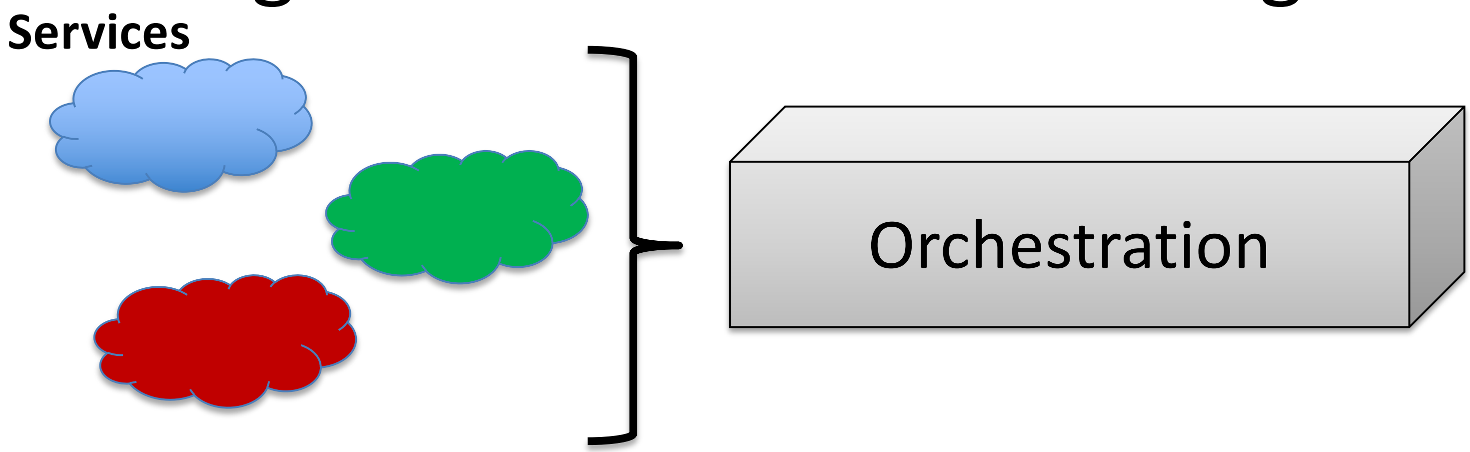
pablo@it.uc3m.es

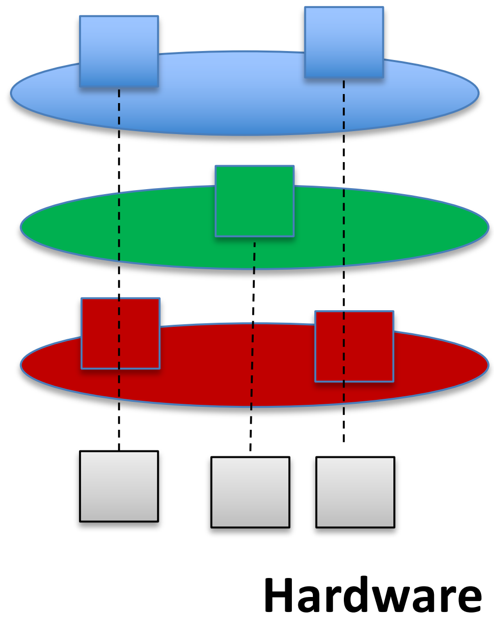
http://www.it.uc3m.es/pablo/

Always looking for motivated people to collaborate with!

Outline of this talk

- Network Slicing supports the instantiation of a logical network tailored to a given service





- Orchestration
  - Expected performance? How to do it?
- Virtualization
  - What are the challenges? How to address them?

Extra: how to deploy your own testbed

WHAT IS NETWORK SLICING?

What is 5G? (2018)

WIKIPEDIA
The Free Encyclopedia

“***5G*** is a marketing term for some new mobile technologies.[*according to whom?*] Definitions differ[*citation needed*] and confusion is common[*citation needed*]. The ITU IMT-2020 standard provides for speeds up to 20 gigabits per second and has only been demonstrated with millimeter waves of 15 gigahertz and higher frequency. The more recent 3GPP standard includes any network using the NR New Radio software.” (Sept. 4th, 2018)

What is 5G? (2019)

WIKIPEDIA
The Free Encyclopedia

“‘**5G**´ is the fifth generation cellular network technology. The industry association 3GPP defines any system using "5G NR" (5G New Radio) software as, "5G", a definition that came into general use by late 2018. Others may reserve the term for systems that meet the requirements of the ITU IMT-2020.” (Oct. 29th, 2019)

What is 5G? (2020)

WIKIPEDIA
The Free Encyclopedia

"In telecommunications, **5G** is the fifth generation technology standard for cellular networks, which cellular phone companies began deploying worldwide in 2019, the planned successor to the 4G networks which provide connectivity to most current cellphones.[1] Like its predecessors, […]

The industry consortium setting standards for 5G is the 3rd Generation Partnership Project (3GPP).[1] It defines any system using 5G NR (5G New Radio) software as "5G", a definition that came into general use by late 2018. Minimum standards are set by the International Telecommunications Union (ITU). Previously, some reserved the term 5G for systems that deliver download speeds of 20 Gbit/s as specified in the ITU's IMT-2020 document." (Sept. 3rd, 2020)

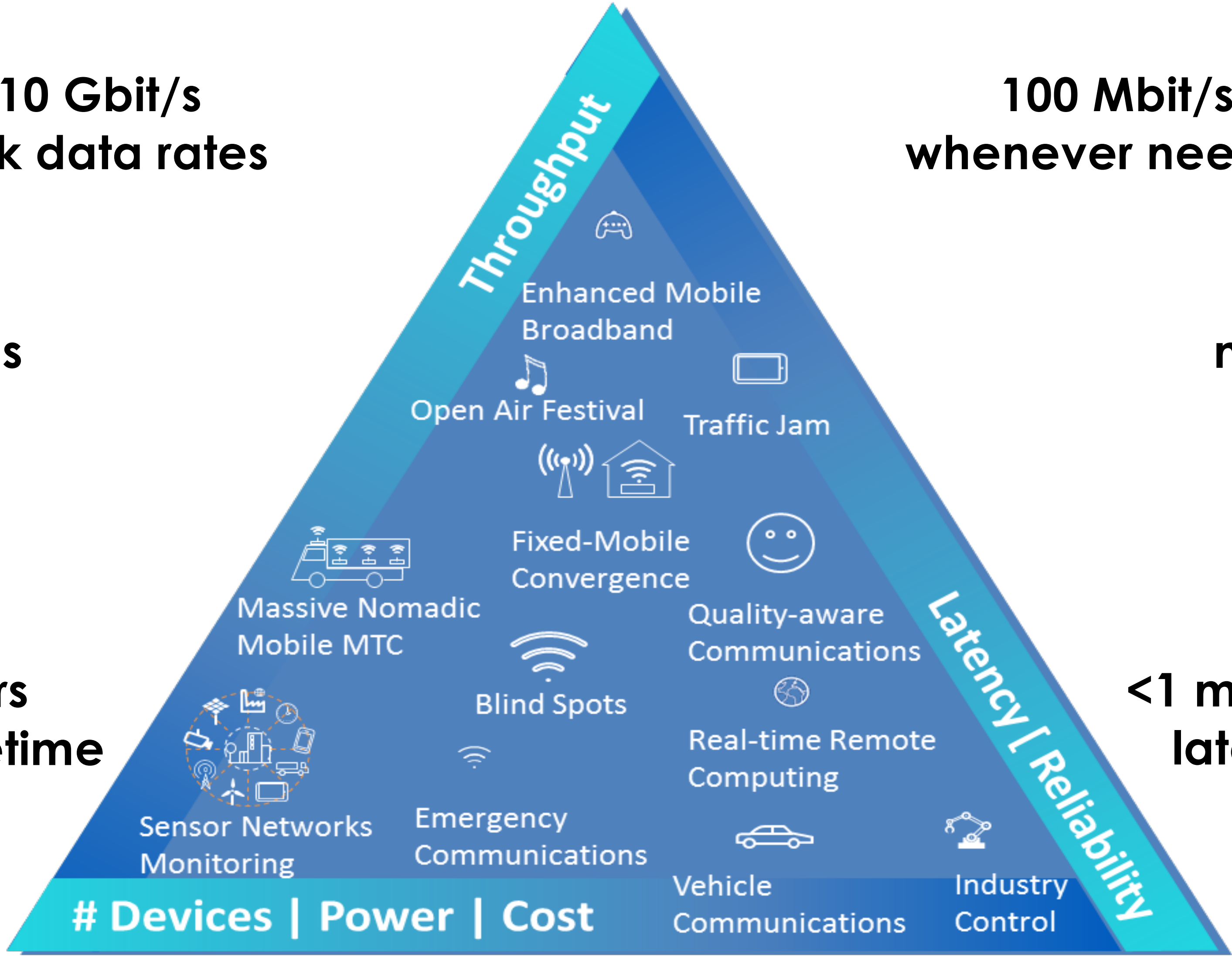
Requirements: Heterogeneity

>10 Gbit/s
peak data rates

10 – 100 x
more devices

M2M
ultra low cost

10 years
battery lifetime



100 Mbit/s
whenever needed

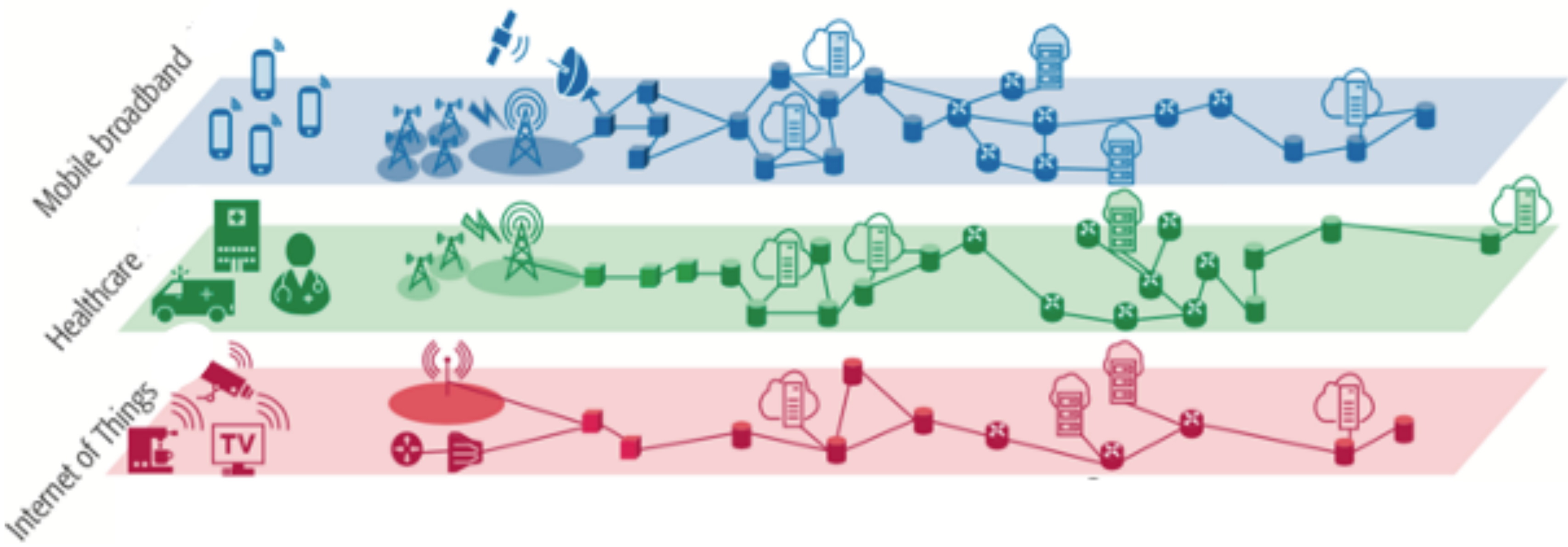
10 000 x
more traffic

ultra
reliability

<1 ms radio
latency

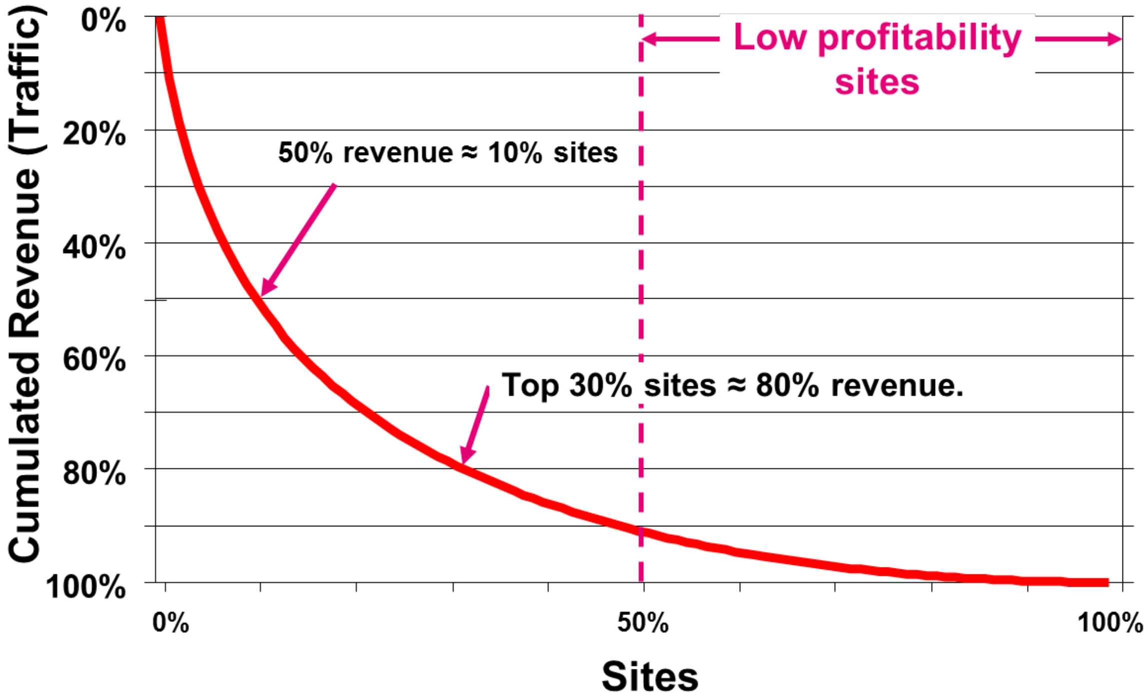
If business as usual…

- One network for all services: does not work
- One network per service: does not scale

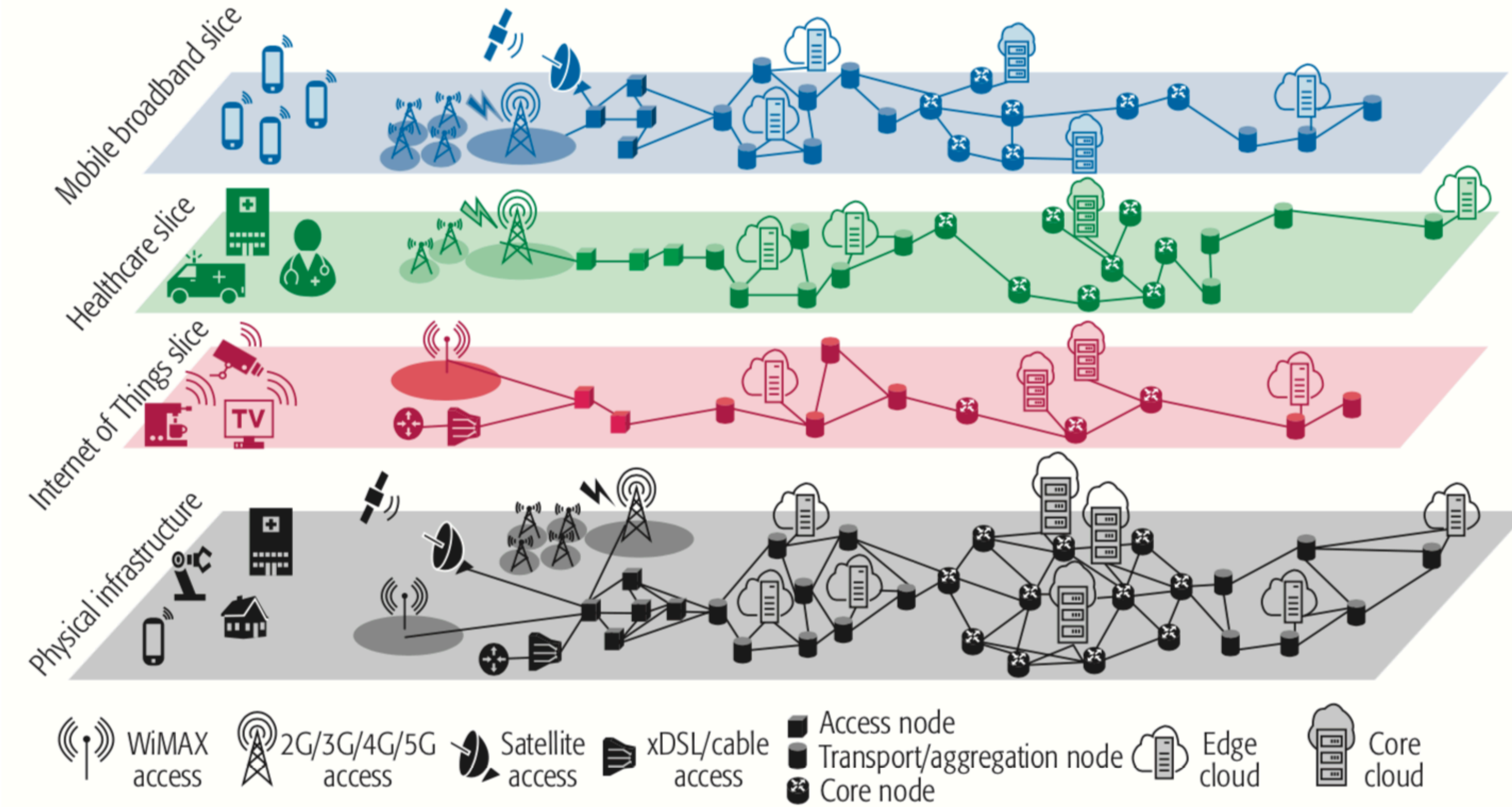


… a lot of inefficiency

- 50% of revenue created by < 10% of sites
- Diverse traffic patterns/mobility cause **resource underutilization**



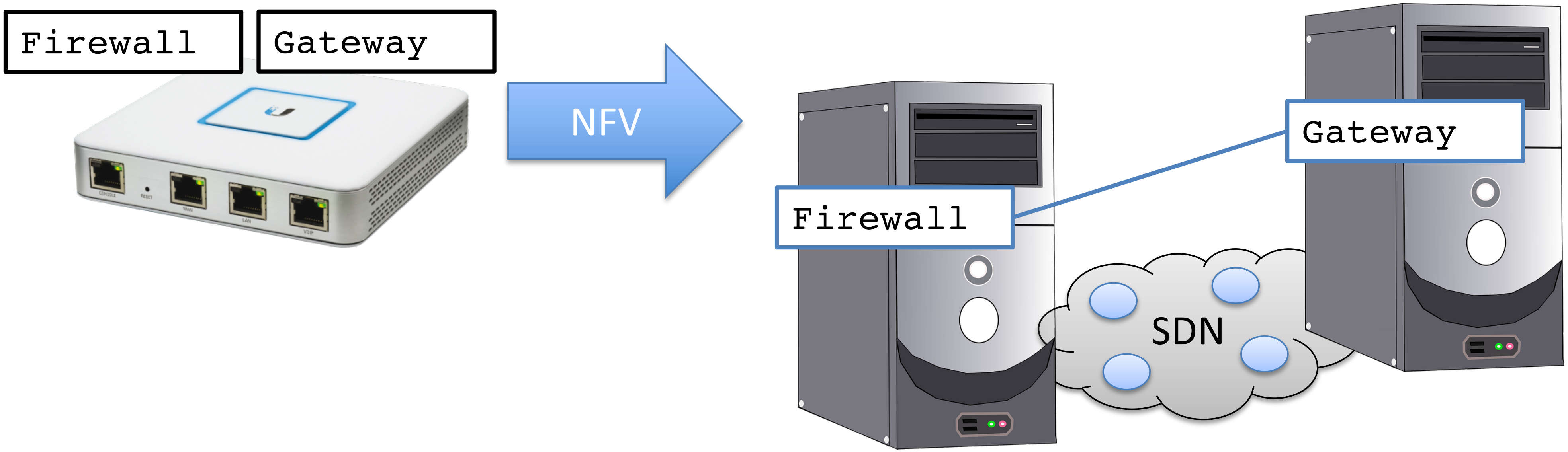
Network Slicing: multiplexing



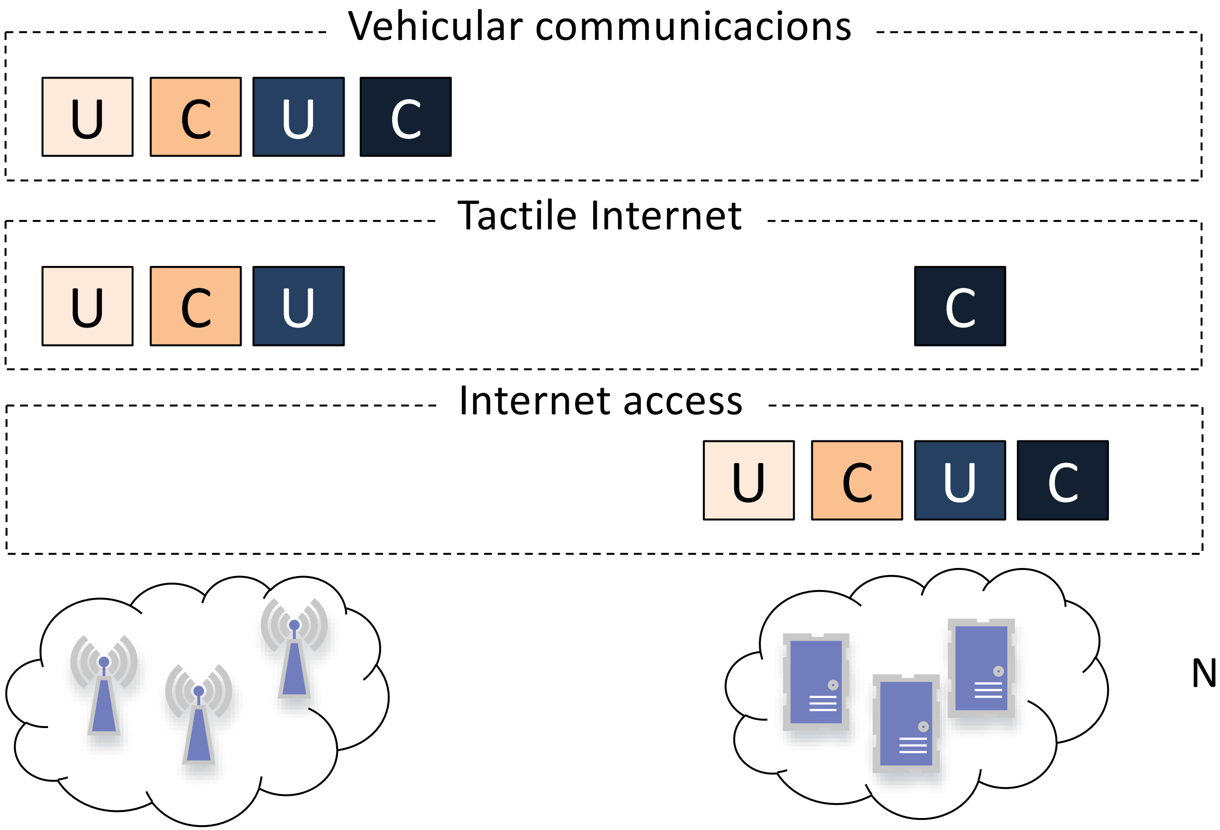
From: J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca and J. Folgueira, "Network Slicing for 5G with SDN/NFV: Concepts, Architectures, and Challenges," in *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80-87, May 2017.

Why now? NFV and SDN

- Virtual Network Functions (VNFs) connected via Software Defined Networking (SDN)



Orchestration of VNFs



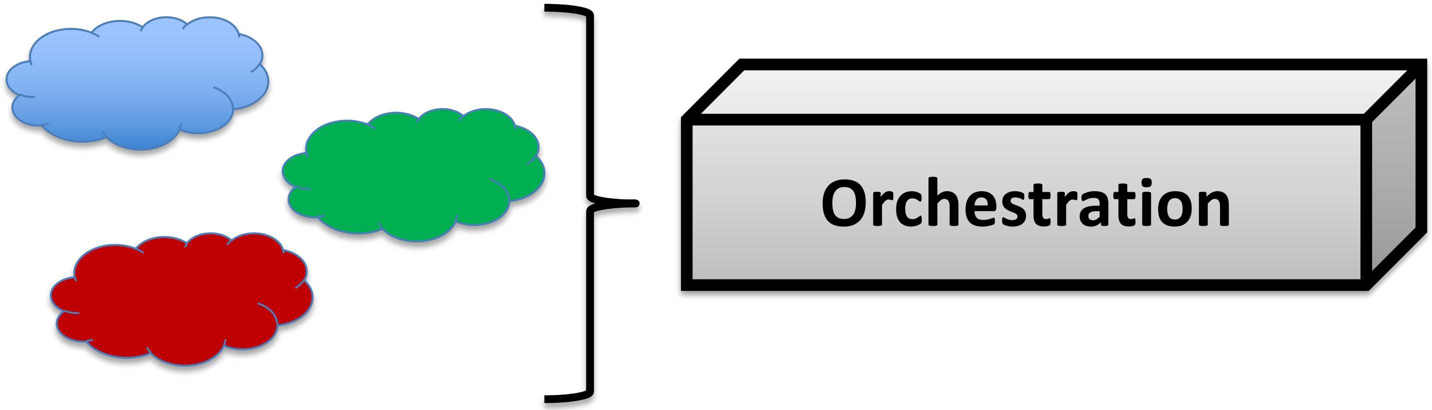
Virtual Network Functions (VNFs)

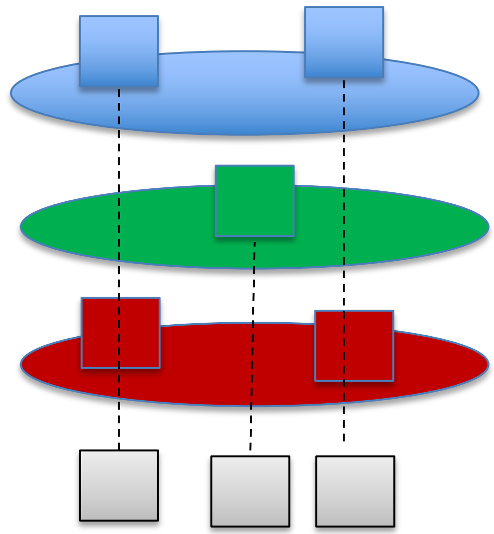
- U U-plane CN
- C C-plane CN
- U U-plane RAN
- C C-plane RAN

Nodes: general purpose processors

Outline

- Network Slicing supports the instantiation of a logical network tailored to a given service

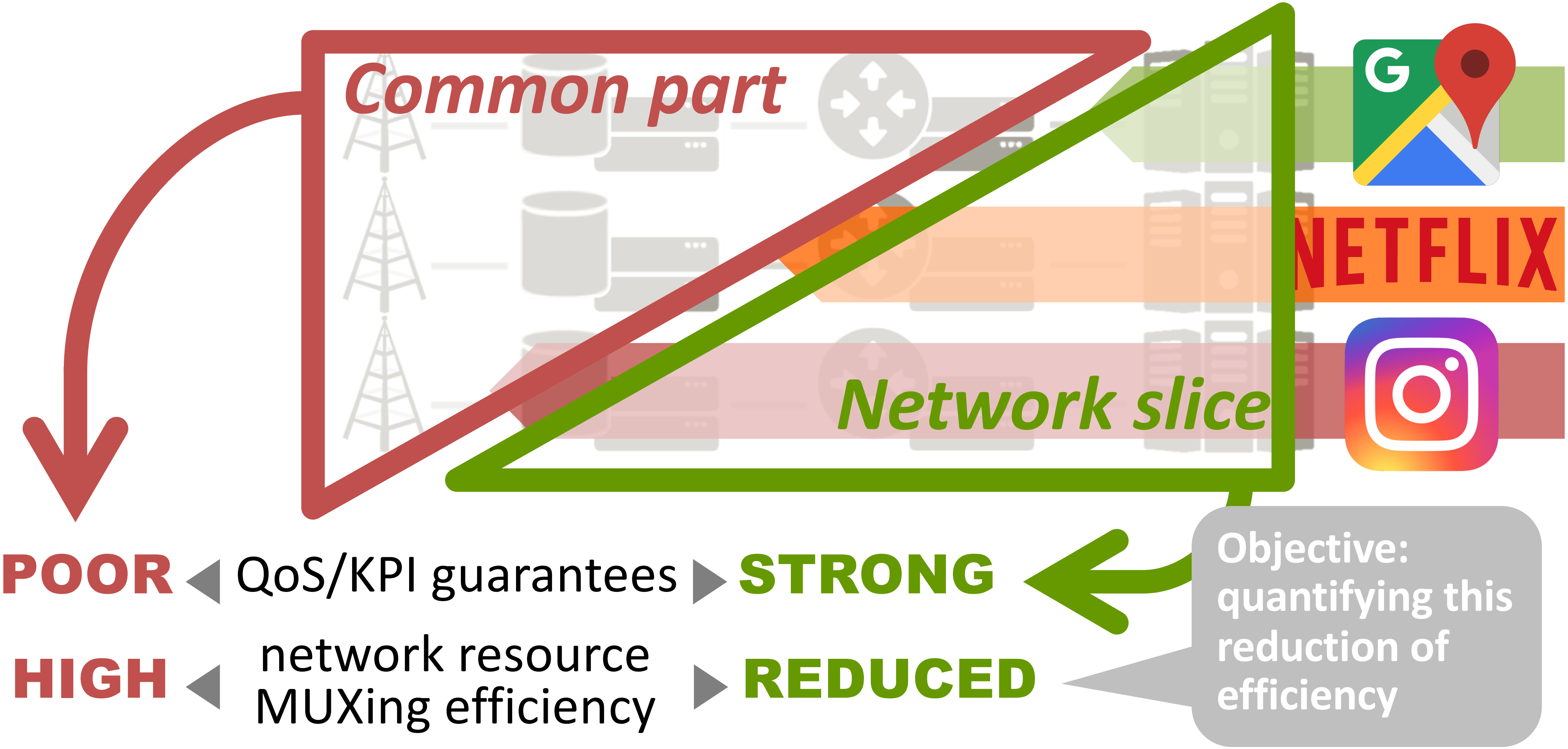




- **Orchestration**
  - **What efficiency is expected? How to do it?**
- Virtualization
  - What are the challenges? How to address them?

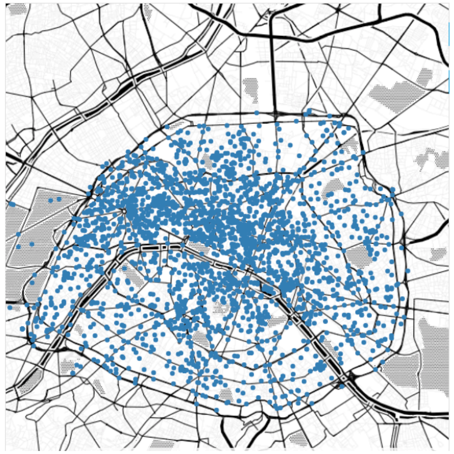
SLICING EFFICIENCY

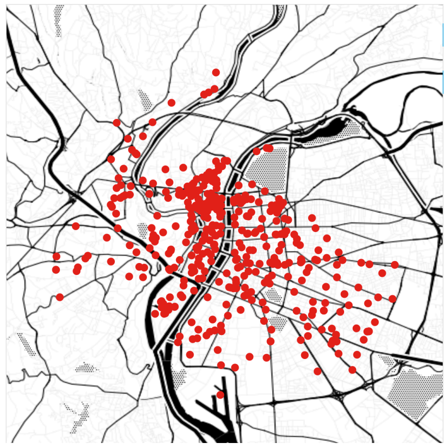
Slicing trade-offs



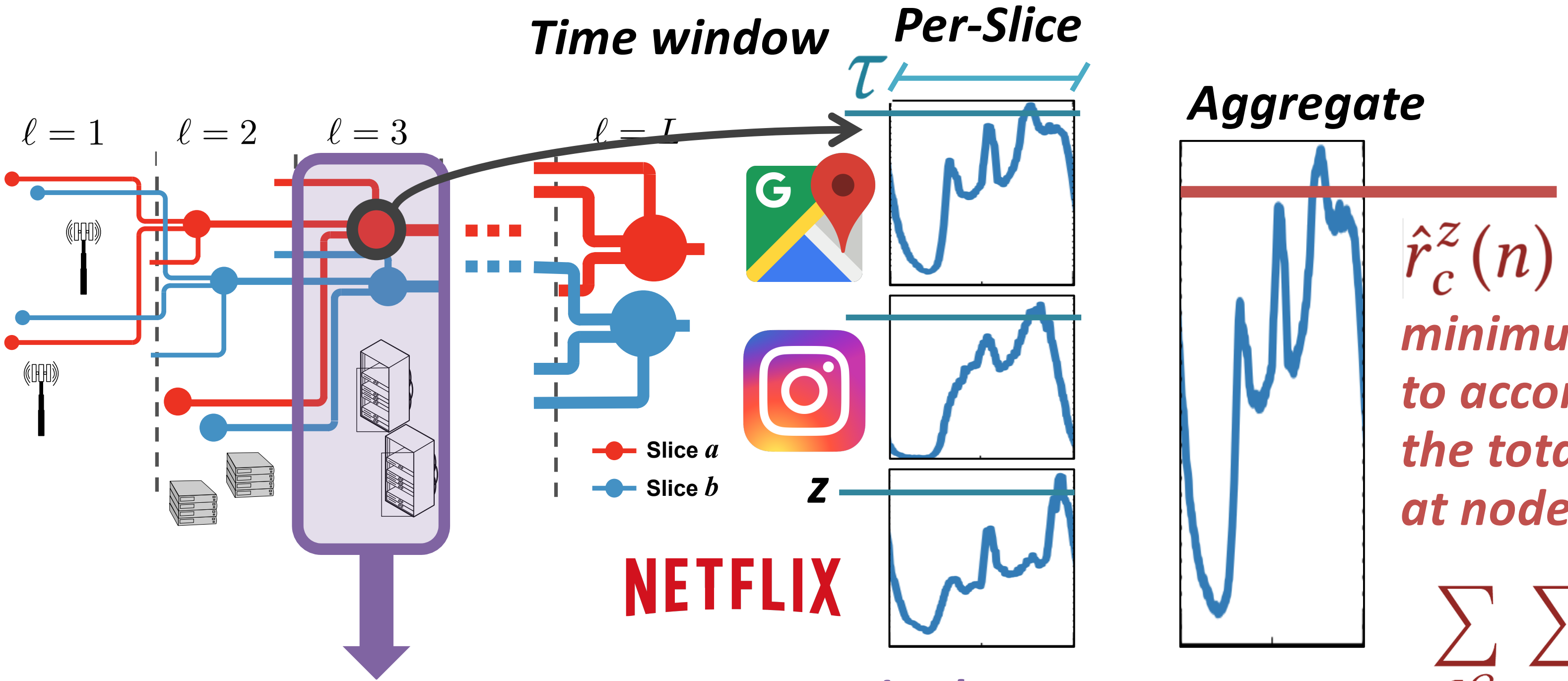
Problem statement

- Analyse **real data** from two cities
- Compare the resources required to serve the traffic
- Using two approaches
  - With slicing
  - No slicing
- At different "slicing depths"
  - From the antenna to the core





Slicing vs. Perfect Sharing



$\hat{r}^z_c(n)$

minimum resources to accommodate the total demand at node c under z

$$\sum_{c \in C_\ell} \sum_{n \in \mathcal{T}} \tau \cdot \hat{r}^z_c(n)$$

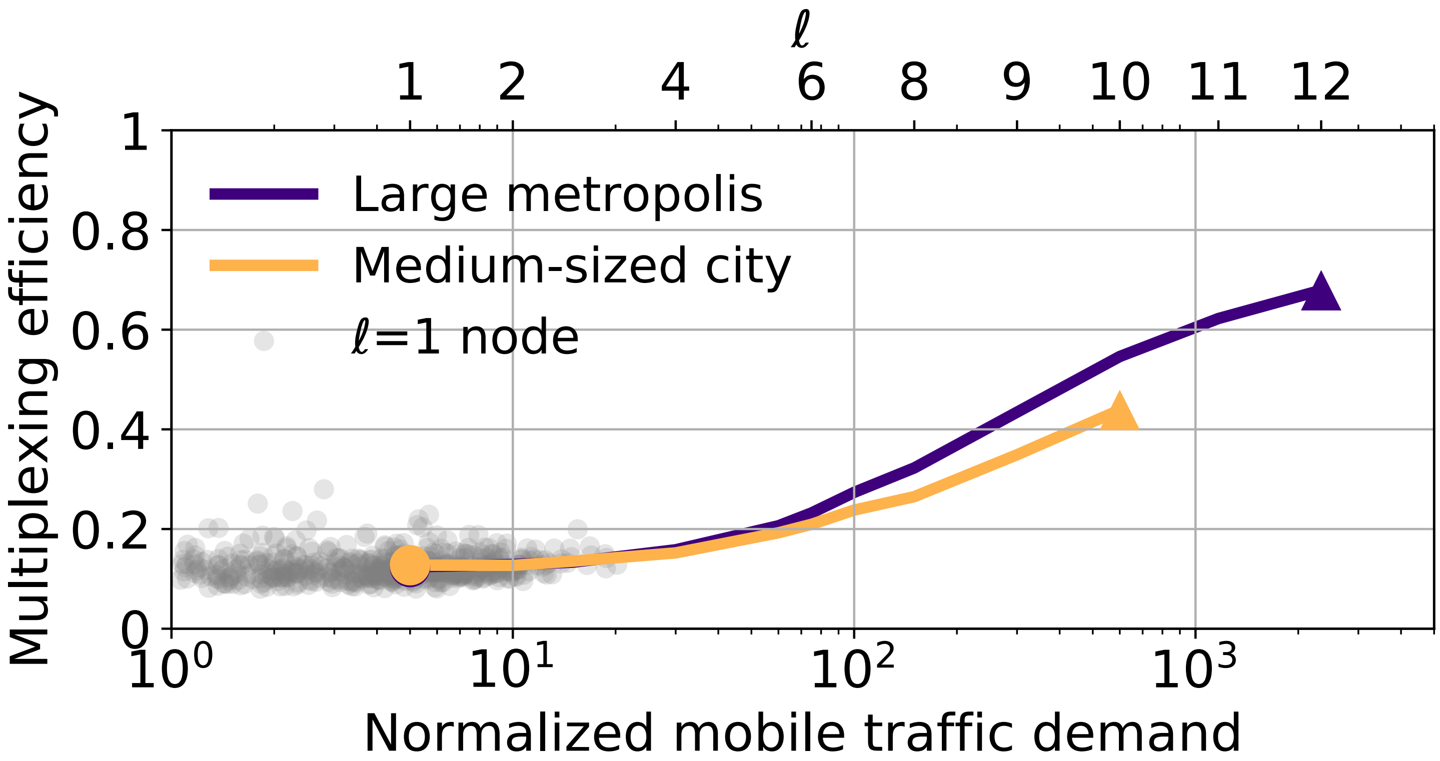
$$\sum_{s \in \mathcal{S}} \sum_{c \in C_\ell} \sum_{n \in \mathcal{T}} \tau \cdot \hat{r}^z_{c,s}(n)$$

resources required to implement slicing at level l under z

Efficiency:

$$\mathbb{R}^z_{\ell,\tau} \Big/ \mathbb{P}^z_{\ell,\tau}$$

Static allocation

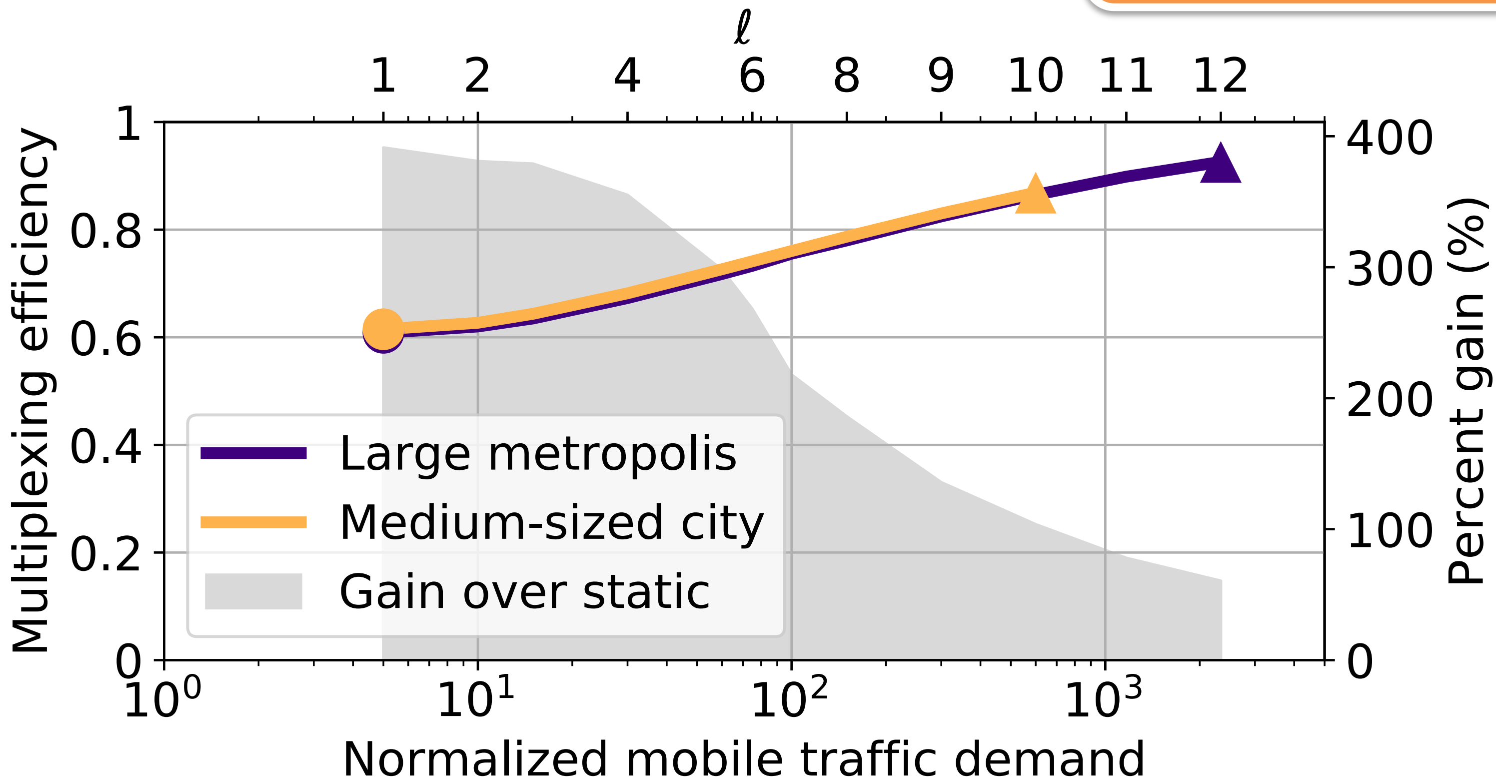


- Resources shall be doubled even at “core depth”

Dynamic allocation

- Re-configuration every 30'

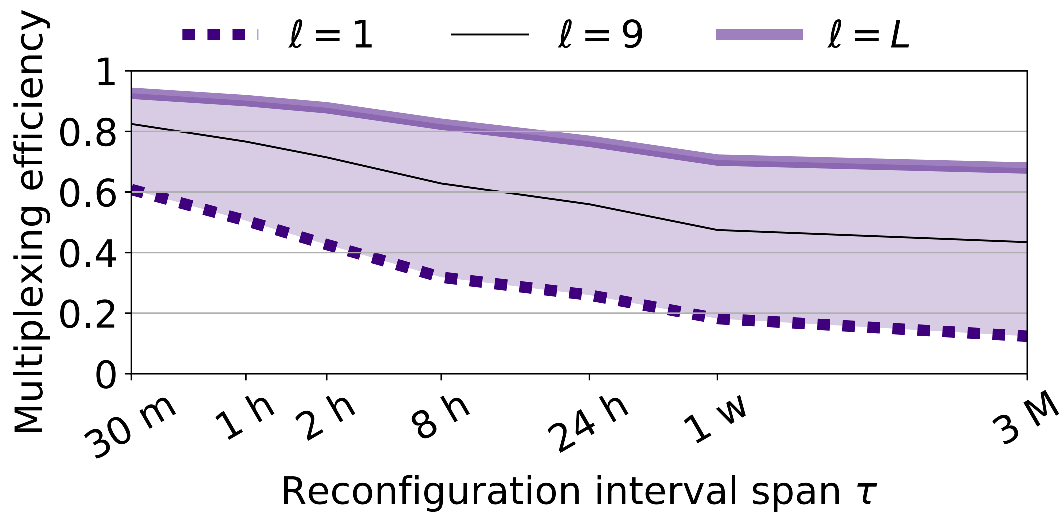
This is not straightforward!



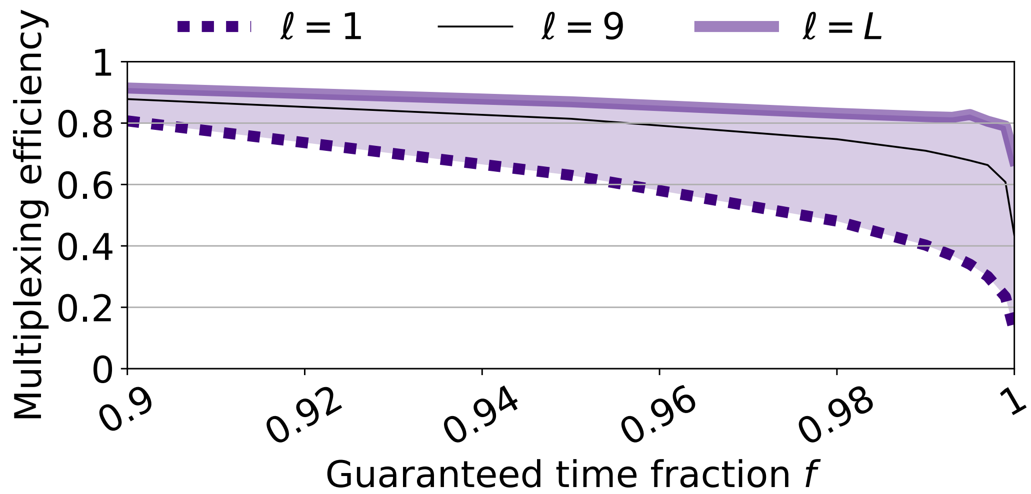
- Up to 4x improvement

Impact of some parameters

- Timescales



- Traffic guarantee

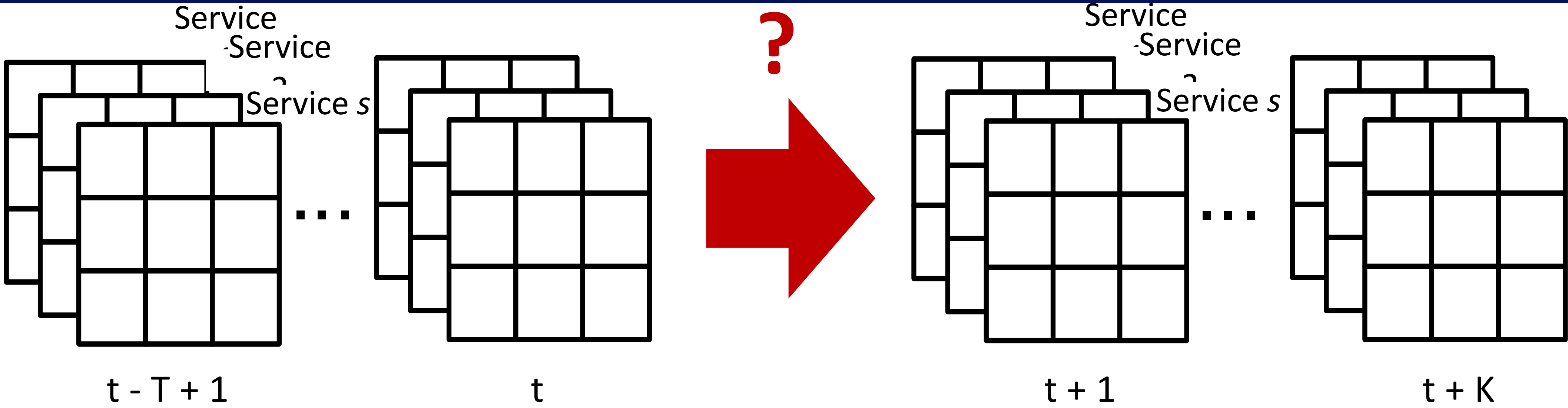


More info

- C. Marquez et al., “How should I slice my network? A multi-service empirical evaluation of resource sharing efficiency,” ACM MobiCom 2018, New Delhi, India

HOW TO PERFORM ORCHESTRATION

The multi-service traffic forecasting problem

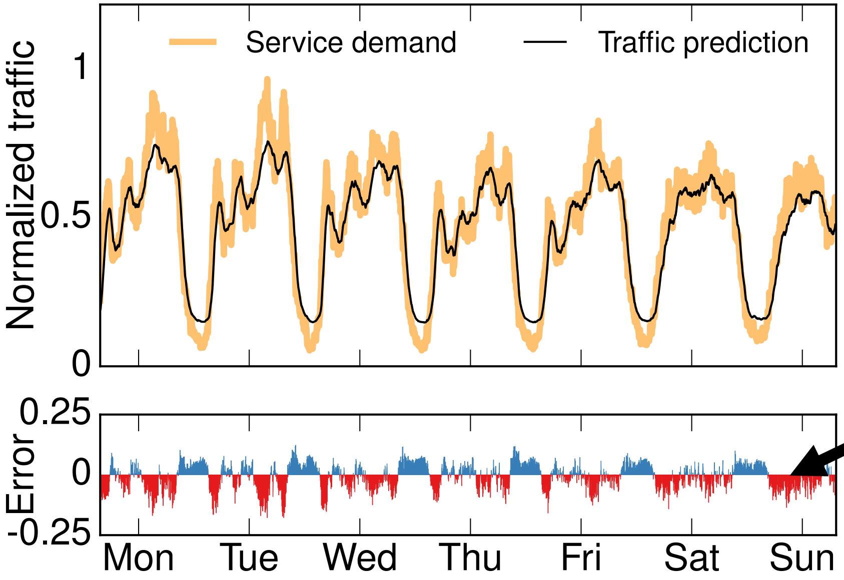


Take historical mobile traffic measurements for multiple services as input and predict future mobile traffic consumption for all services at all base stations simultaneously.

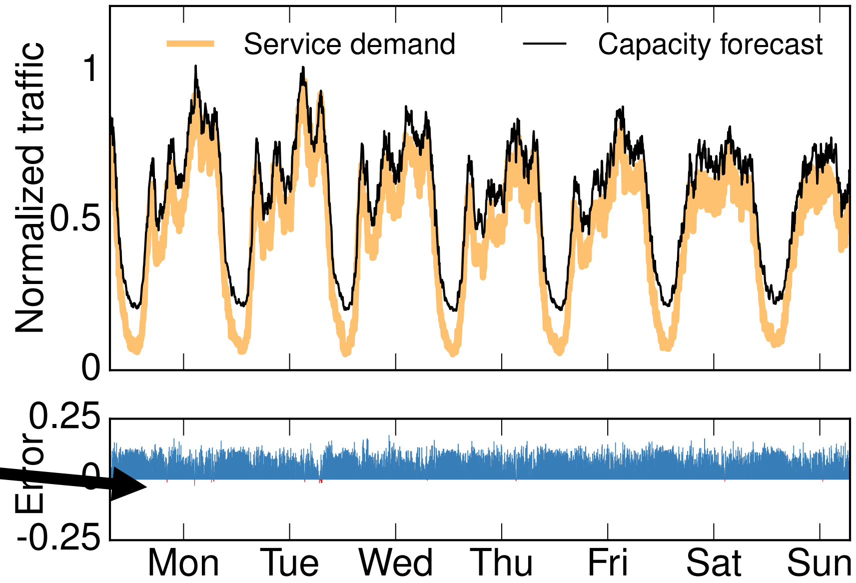
Traditional methods (ES, ARIMA) are inappropriate: Operate on individual time series, their performance degenerates considerably over time, and do not exploit correlations between different services.

Plus: capacity vs. traffic

- Traditional approaches deal with **demand** forecasting

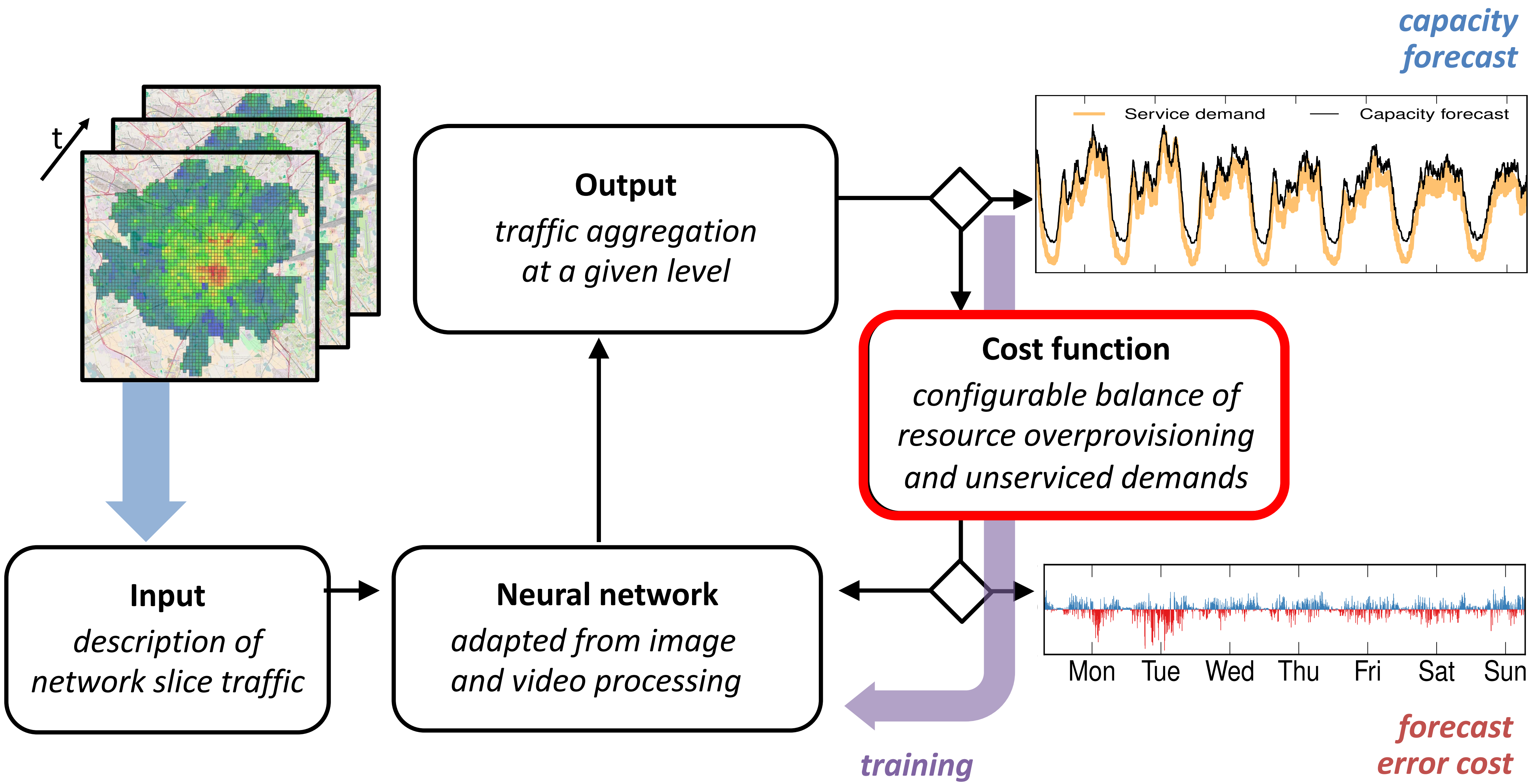


A traffic demand forecasting algorithm aims to minimize the error wrt to the original data, so underestimation is possible



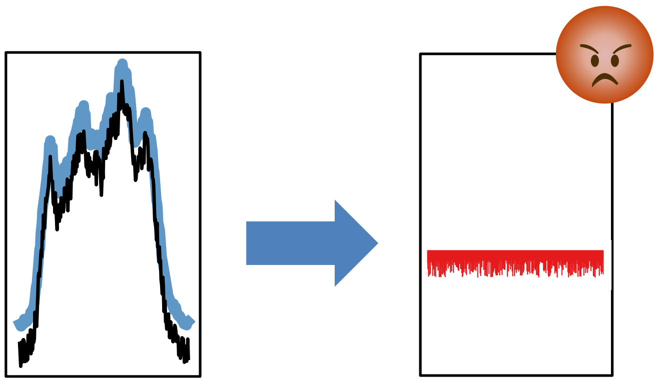
A capacity forecasting algorithm minimizes the amount of resources needed to serve a given demand

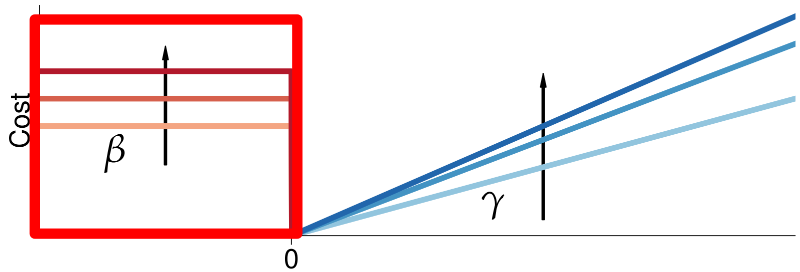
DeepCog


capacity forecast
Service demand
Capacity forecast
t
Output
traffic aggregation at a given level
Cost function
configurable balance of resource overprovisioning and unserviced demands
Input
description of network slice traffic
Neural network
adapted from image and video processing
Mon
Tue
Wed
Thu
Fri
Sat
Sun
forecast error cost
training

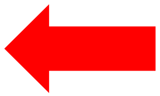
Cost function

- Determines the penalty incurred when making an error
- Tailored to capacity forecast problem
- It accounts for the costs resulting from
  - SLA violations
  - Overprovisioning



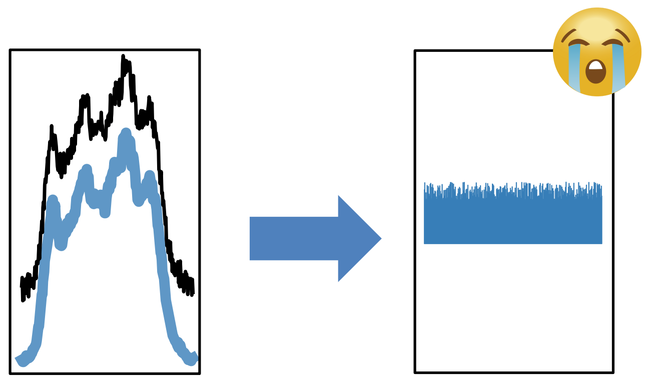


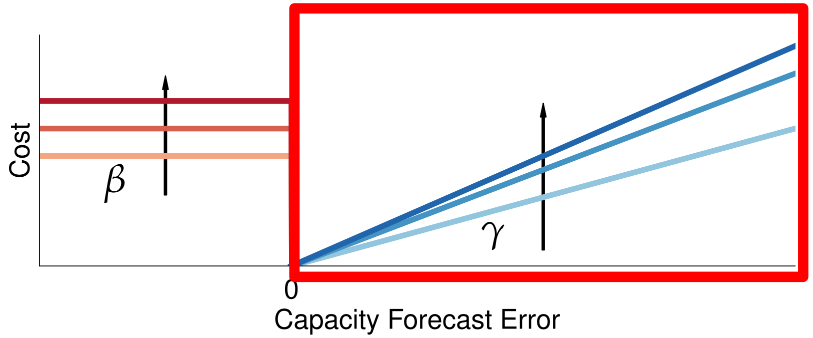
$$\ell(x) = \begin{cases} \beta & \text{if } x \leq 0 \\ \gamma \cdot x & \text{if } x > 0 \end{cases}$$



Cost function

- Determines the penalty incurred when making an error
- Tailored to capacity forecast problem
- It accounts for the costs resulting from
  - SLA violations
  - Overprovisioning

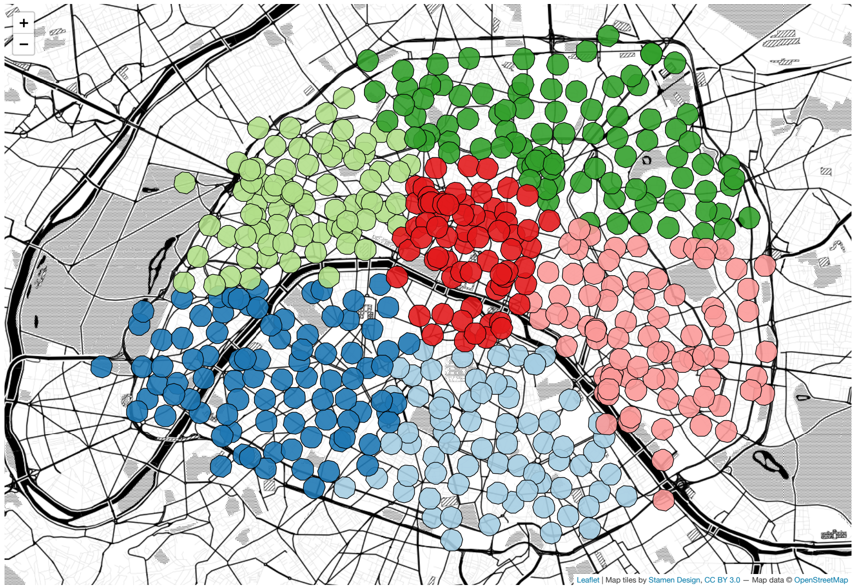




$$\ell(x) = \begin{cases} \beta & \text{if } x \leq 0 \\ \gamma \cdot x & \text{if } x > 0 \end{cases}$$

Performance Evaluation

- Real-world demand generated by several millions of users
  - Mobile network deployed in a large city
- Different services analysed. Core, MEC, & C-RAN

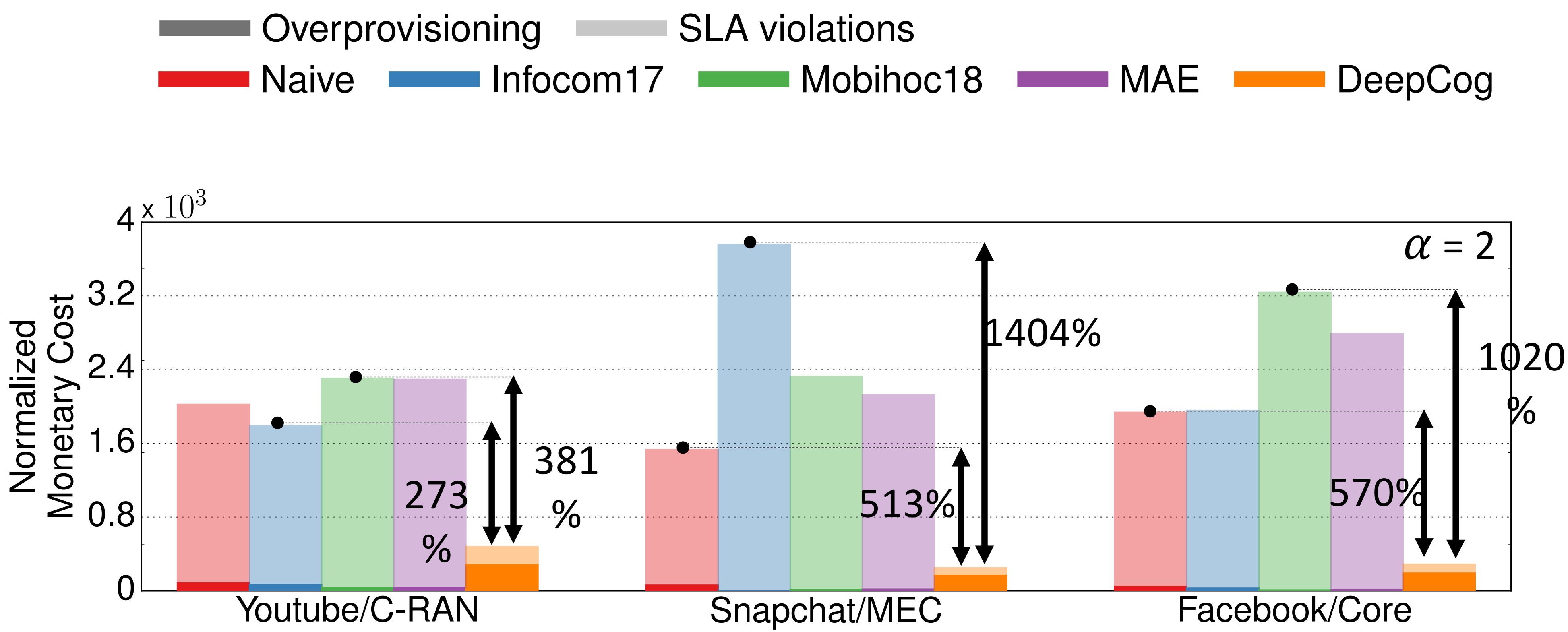


Results - Benchmarks

- Compared against 4 benchmarks
  - Naïve – replicates the previous week demand
  - Infocom17[1] – first DL approach
  - MobiHoc18[2] – recent DL solution
  - MAE – DeepCog architecture employing Mean Absolute Error as loss function

[1] – J. Wang et al., *"Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach,"* in *Proc. of IEEE Infocom, May 2017.*
[2] – C. Zhang et al., *"Long-Term mobile traffic forecasting using Deep Spatio-Temporal Neural Networks,"* in *Proc. Of ACM MobiHoc,* Jun. 2018.

Results



Overprovisioning
SLA violations
Naive
Infocom17
Mobihoc18
MAE
DeepCog
x 10^3
4
3.2
2.4
1.6
0.8
0
Normalized
Monetary Cost
α = 2
273
%
381
%
513%
1404%
570%
1020
%
Youtube/C-RAN
Snapchat/MEC
Facebook/Core

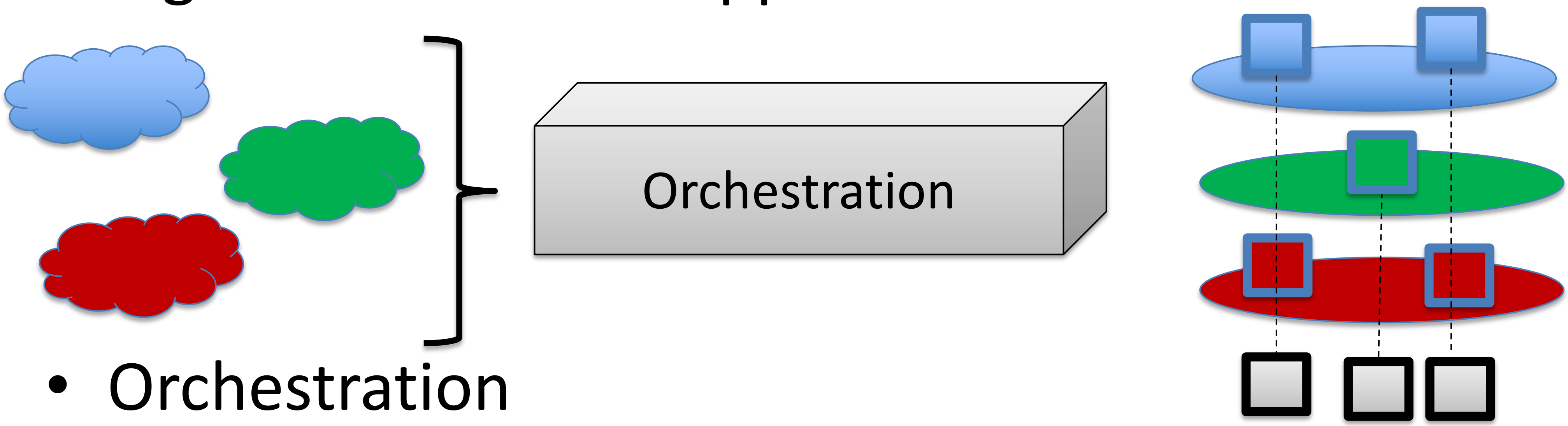
More info & Results

- D. Bega et al., “DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning,” IEEE INFOCOM, April 2019

- C. Zhang et al. “Multi-Service Mobile Traffic Forecasting via Convolutional Long Short-Term Memories,” IEEE International Symposium on Measurements & Networking (M&N), July 2019

Outline

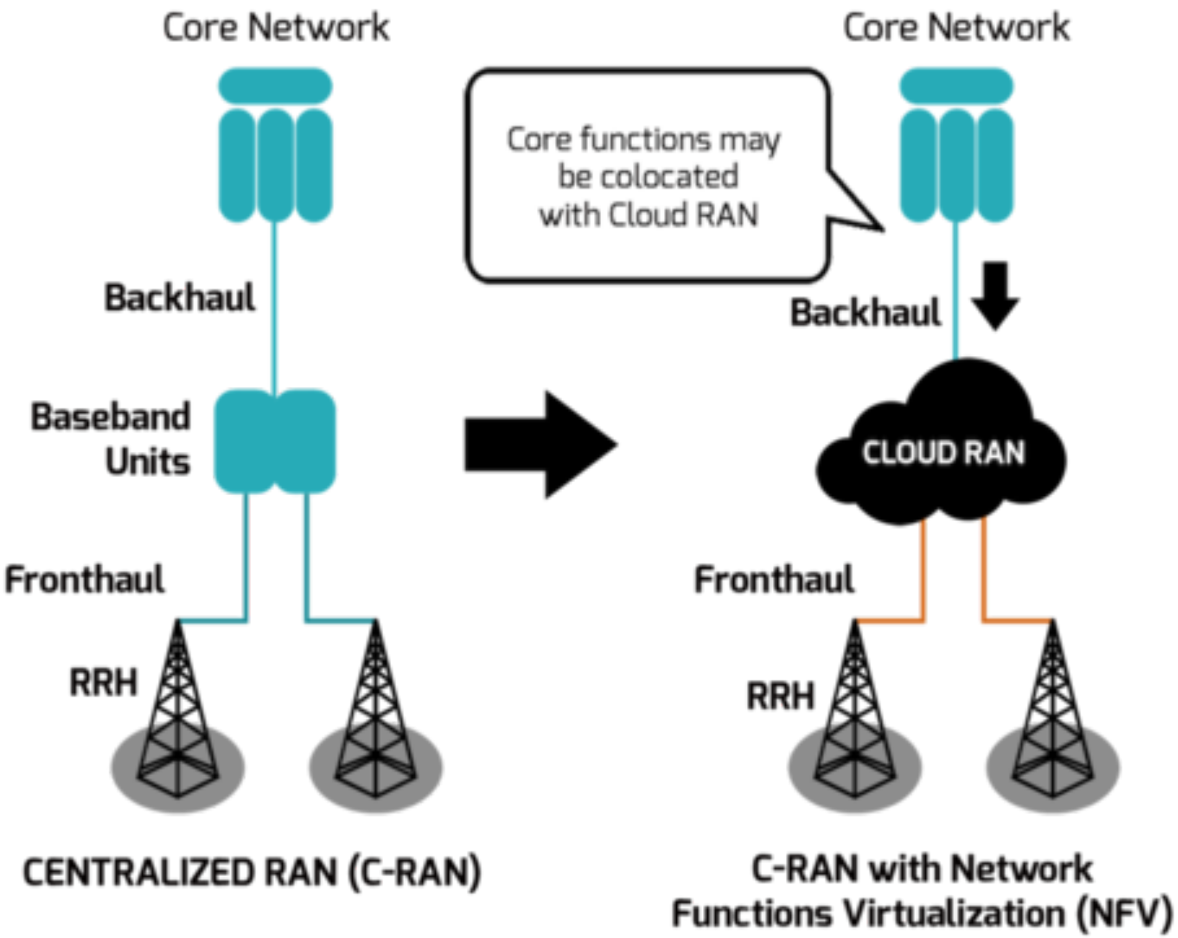
- Network Slicing supports the instantiation of a logical network to support a service



- Orchestration
  - What gains are expected? How to do it?
- **Virtualization**
  - **What are the challenges? How to address them?**

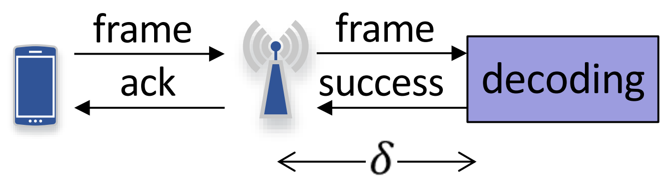
ADDRESSING THE VIRTUALIZATION OF NETWORK FUNCTIONS

Case Study: from C-RAN to vRAN



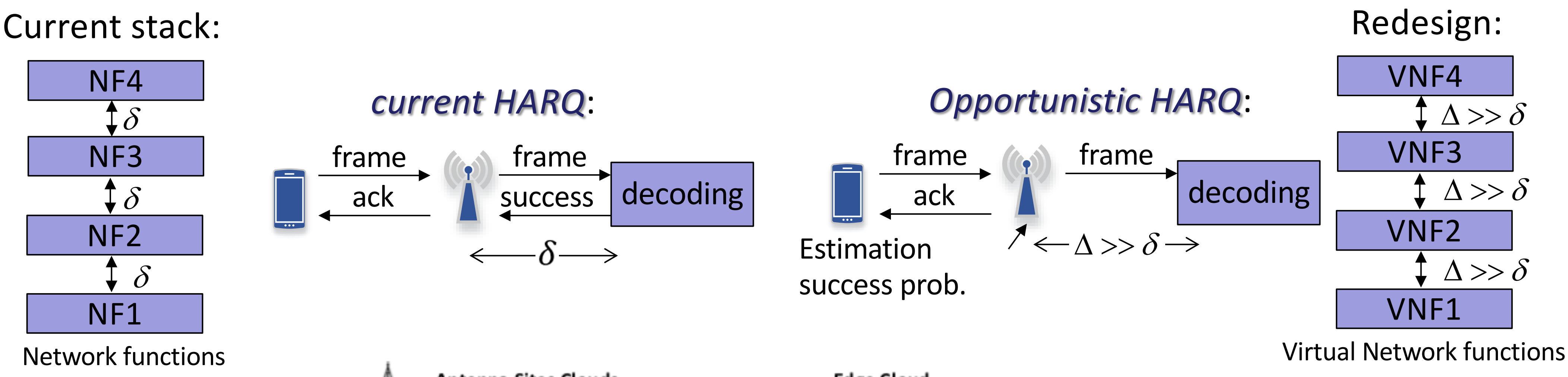
Network functions in the cloud

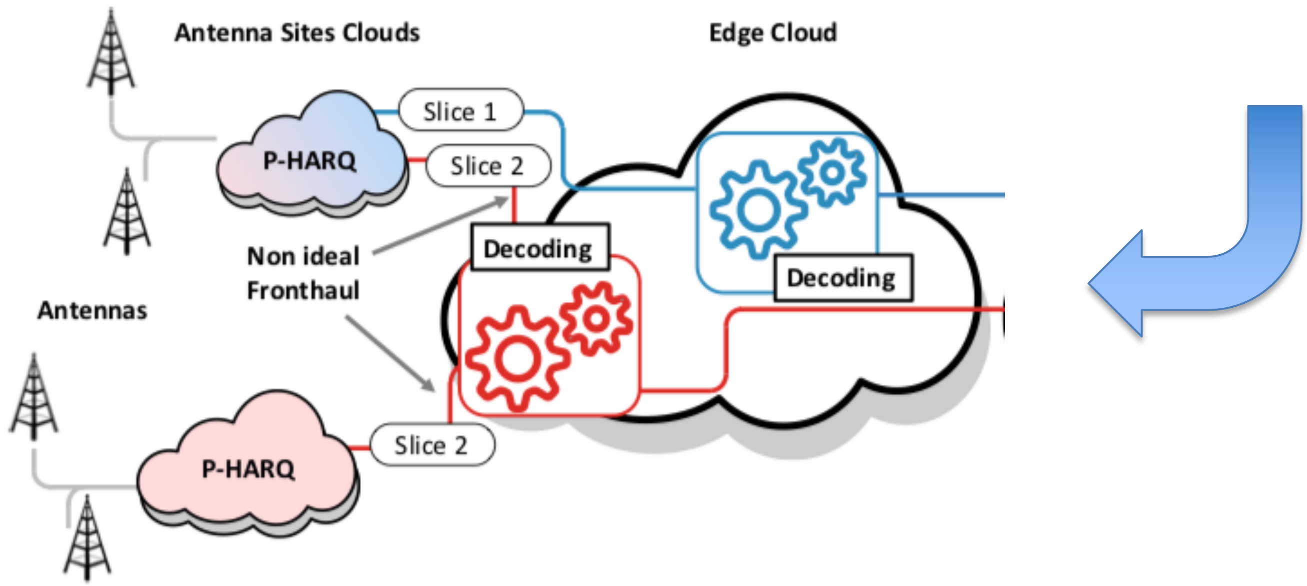
- Traditional approach: dimension for maximum capacity (i.e., all users using highest MCS)
  - E.g., 3 cells, max. 20 users/cell -> 60 users at max. MCS
  - Resource wastage if this rarely happens
- But some functions do not tolerate resource shortage (e.g., decoding): they are *inelastic*



- No resources to decode -> frame lost

1) Removing tight constrains

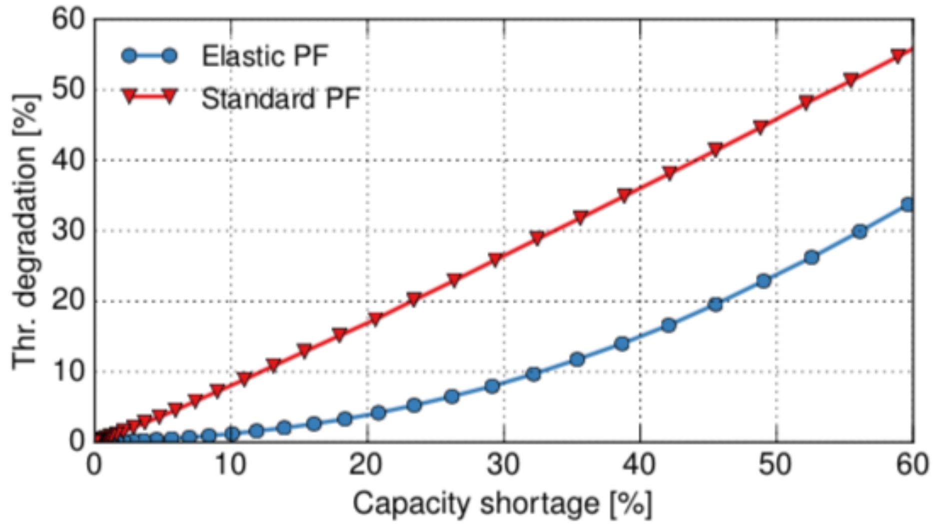




P. Rost and A. Prasad, "Opportunistic Hybrid ARQ – enabler of centralized-ran over non-ideal backhaul," IEEE Wir. Comm. Mag

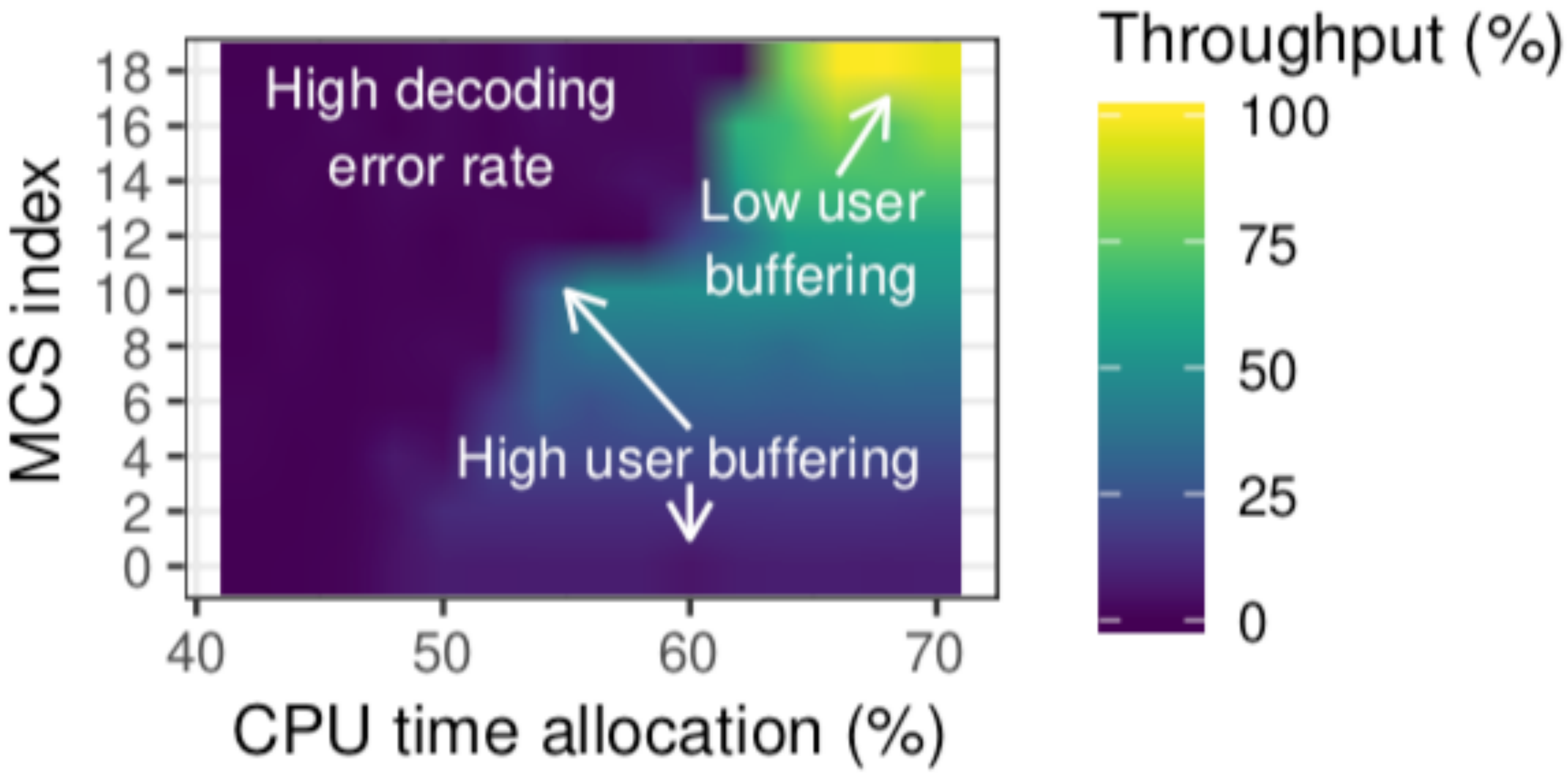
2) Rethinking network functions

- Different approach: assume there could be capacity shortage
- “There is no CPU for this MCS, but we could schedule this other (lower) MCS”
- Resource-elastic schedulers

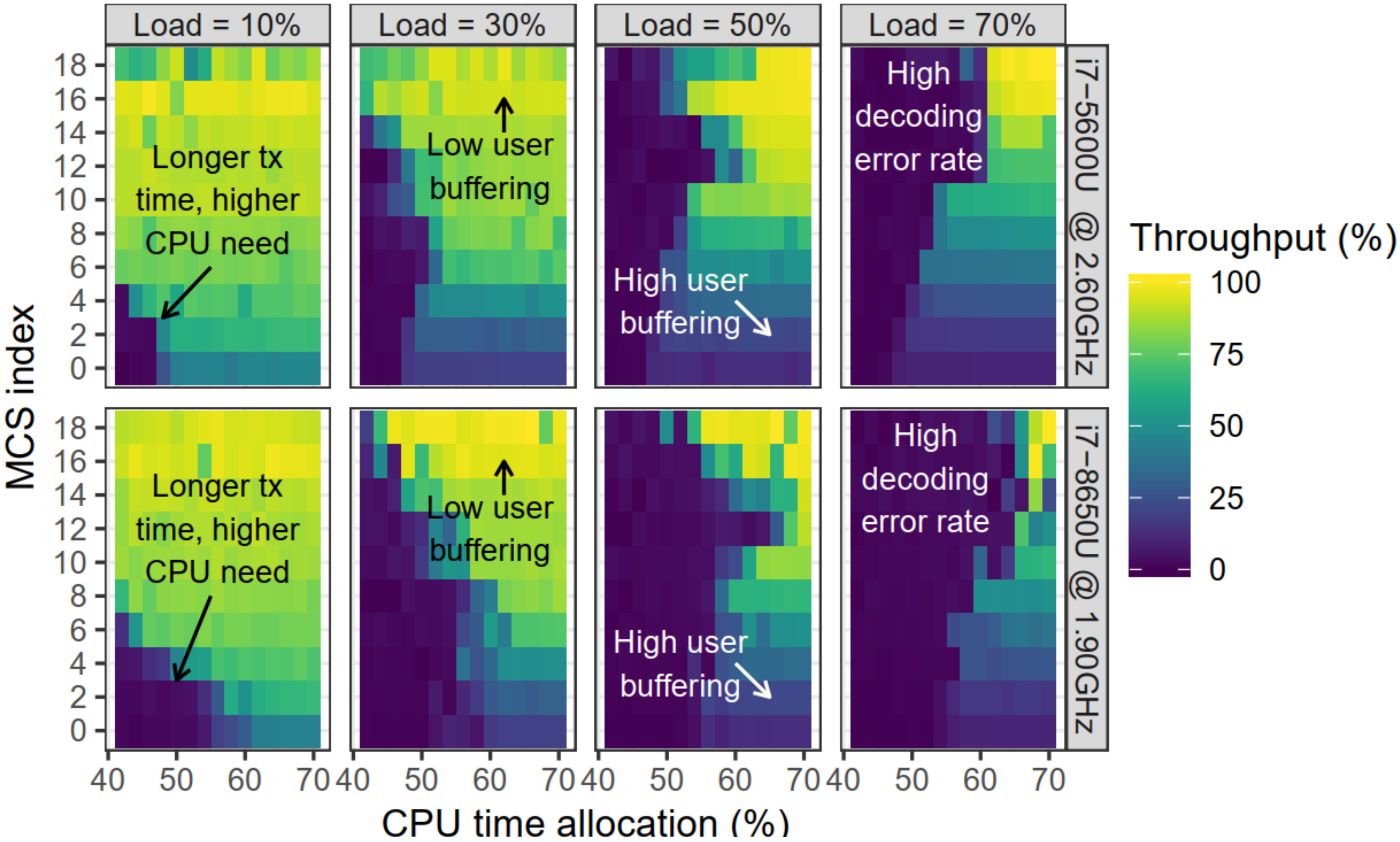


Relation between parameters

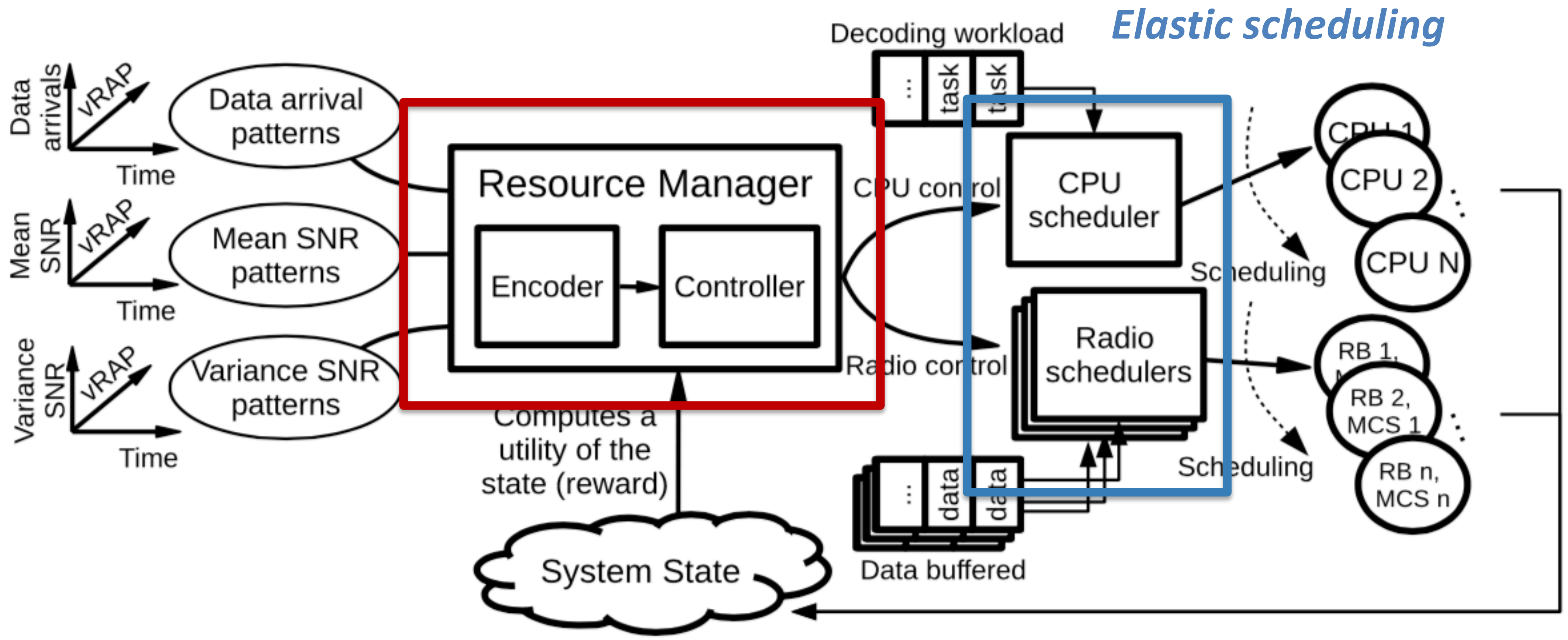
- Throughput as a function of MCS index and CPU time allocation



Relation is complex and non-linear

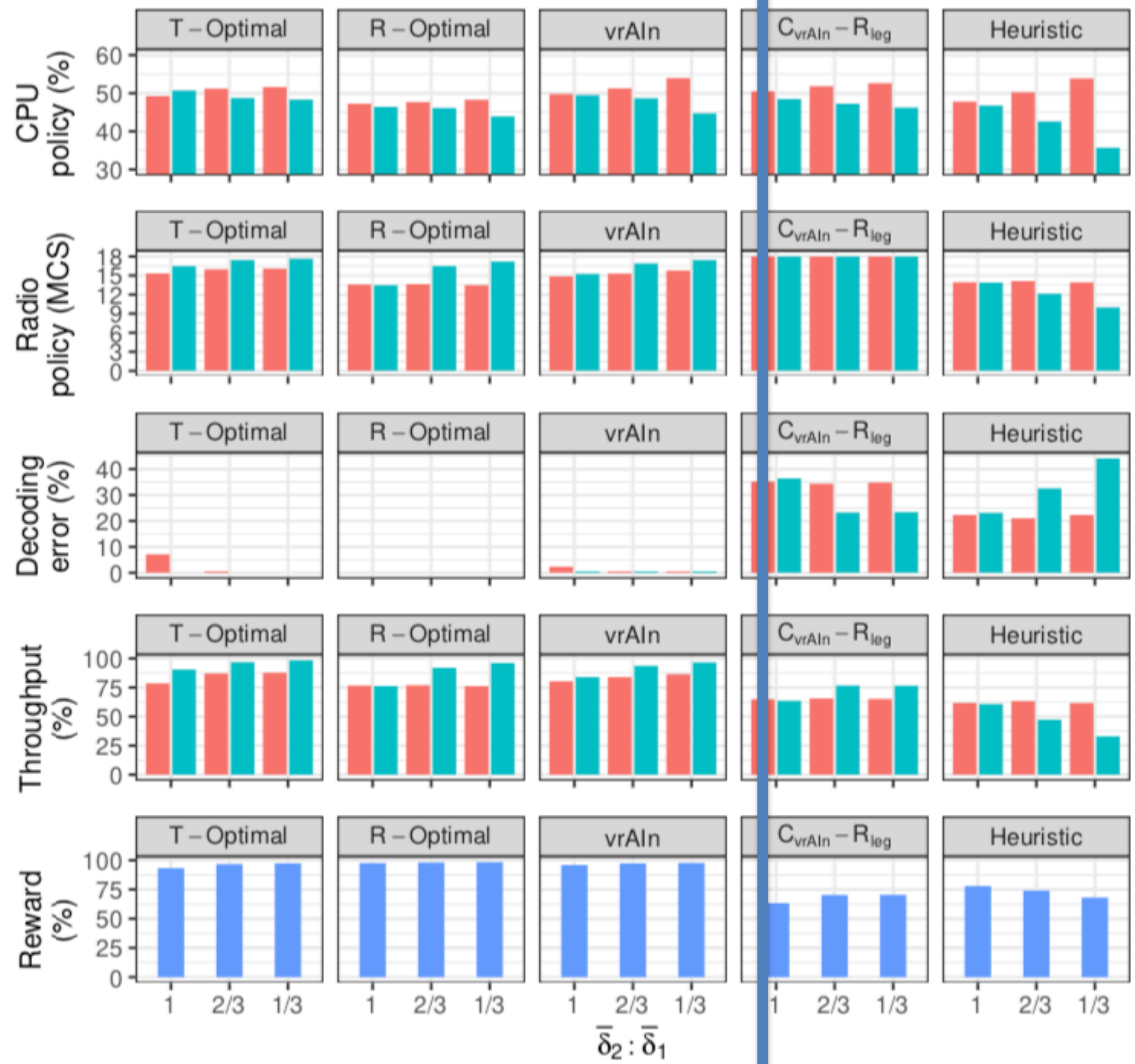


Machine Learning



Results

- Throughput-Optimal oracle
- Reward-optimal oracle,
- **vrAIn**
- Legacy radio
- Heuristic



More info

- P. Serrano et al., “The path towards a cloud-aware mobile network protocol stack,” Transactions on E.T.T., May 2018
- F. Gringoli et al., “Performance Assessment of Open Software Platforms for 5G Prototyping,” IEEE Wireless Communications Magazine, Special Issue on 5G Testing and Field Trials
- J. A. Ayala-Romero et al., “vrAIn: A Deep Learning Approach Tailoring Computing and Radio Resources in Virtualized RANs,” ACM Mobicom, 2019
- M. Gramaglia et al. “The case for sustainable serverless networking,” working paper

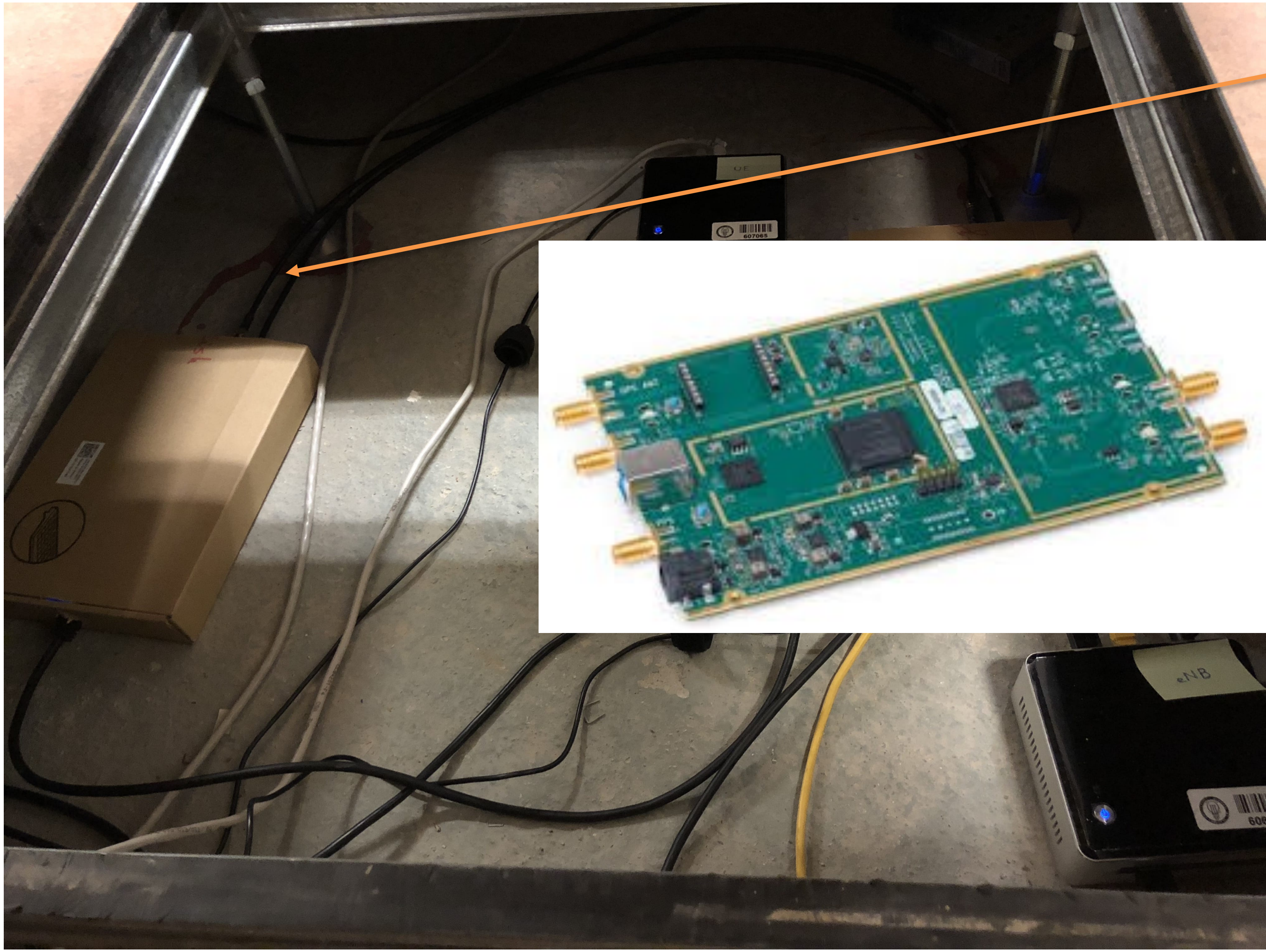
PROTOTYPE DEPLOYMENT

From: G. García-Avilés et al., “POSENS: a Practical Open-source Solution for End-to-end Network Slicing,” IEEE Wir. Comm. Mag.

Elements

- Infrastructure
  - Radio Access Network and Core Network
- Virtual Infrastructure manager (VIM)
  - To run and connect virtual machines
- Network functions
- Orchestrator
  - The ability to instantiate, run, assign resources, monitor performance, etc.

RAN - SDR



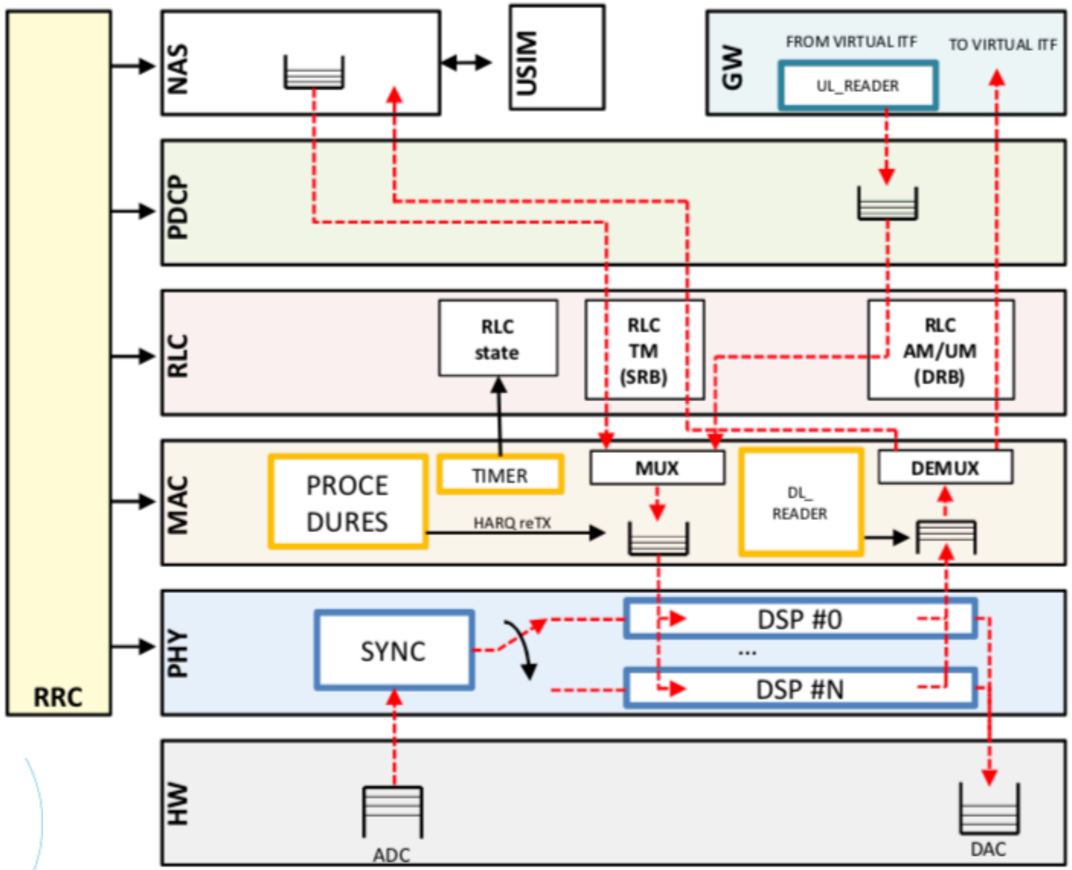
Wires

Ettus B210

Intel NUC

RAN - Functions

- **Software** Defined Radio
  - Many components are implemented by software on a personal computer
  - “Affordable” for Academia
  - “Open” projects
- Two prevalent choices
  - srsLTE
  - OpenAirInterface



From: Gomez-Miguelez et al., “srsLTE: An Open-Source Platform for LTE Evolution and Experimentation,” ACM WiNTECH 2016

Software pitfalls

- Bandwidth incompatibilities:
  - SRS supports operation with all the bandwidth settings specified by 3GPP, i.e., 1.4, 3, 5, 10, 15, and 20 MHz, OAI does not work with the 1.4, 3 and 15 MHz configurations
  - srsENB implementation does not work reliably with 20MHz channels
- Interconnection with CN
  - srsENB employs the same subnetwork for both user plane (S1-U interface) and control plane (S1-C interface).
  - OpenAir-CN can be configured to use two different subnetworks
- Problematic queue management

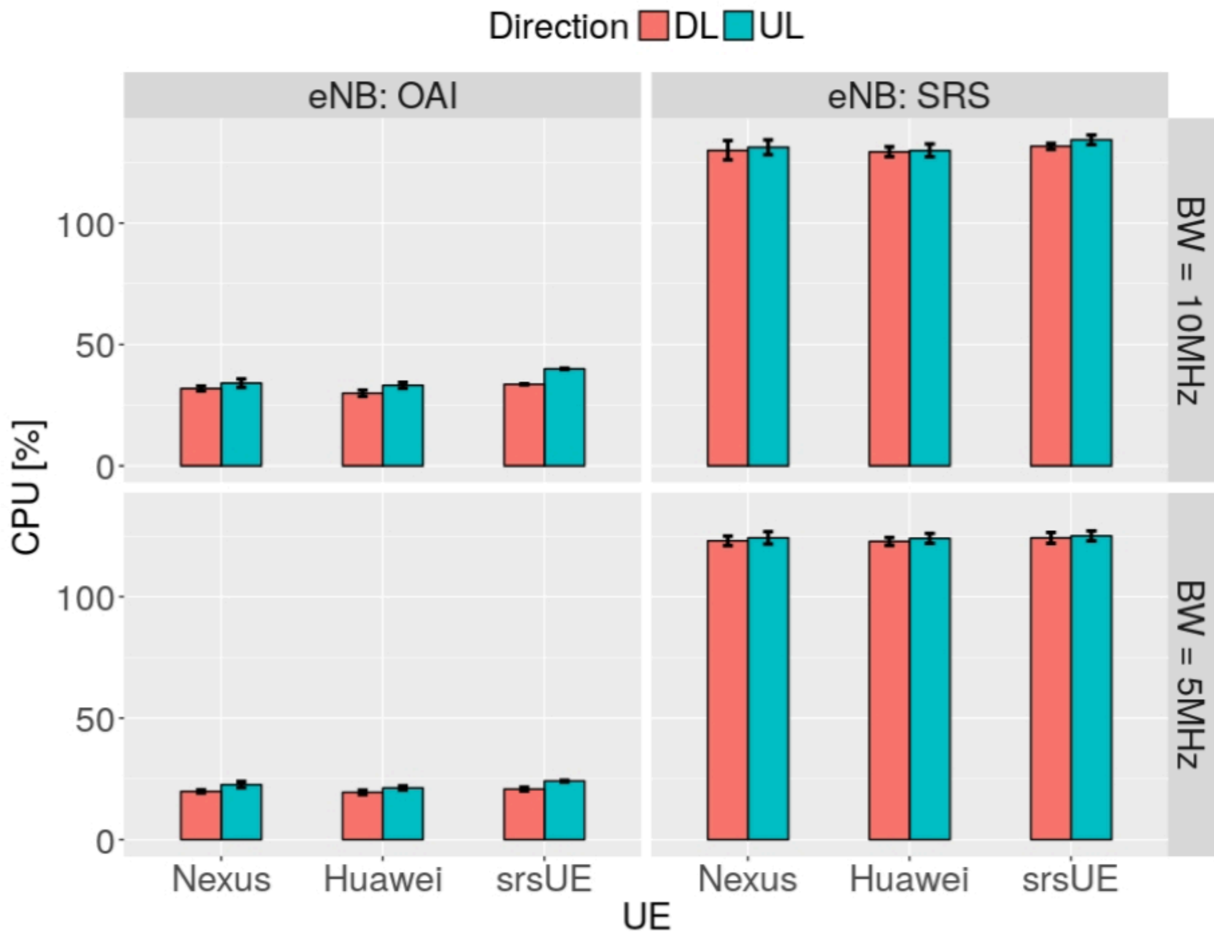
From: F. Gringoli et al., "Performance Assessment of Open Software Platforms for 5G Prototyping," IEEE Wir. Comm. Mag.

SRS vs OAI

- srsLTE's more modular and easier to customize
- For instance: fix during experiments the MCS assignments which the eNB enforces
  - **srsLTE**: function `sched_ue` in `srsLTE/srsenb/src/mac/scheduler_ue.cc`
  - **OAI**: files `eNB_scheduler_ulsch.c` and `eNB_scheduler_dlsch.c` contain the functions `schedule_ulsch_rnti` and `schedule_dlsch_rnti`
    - We developed a patch
    - We discovered that it was altered by other functions

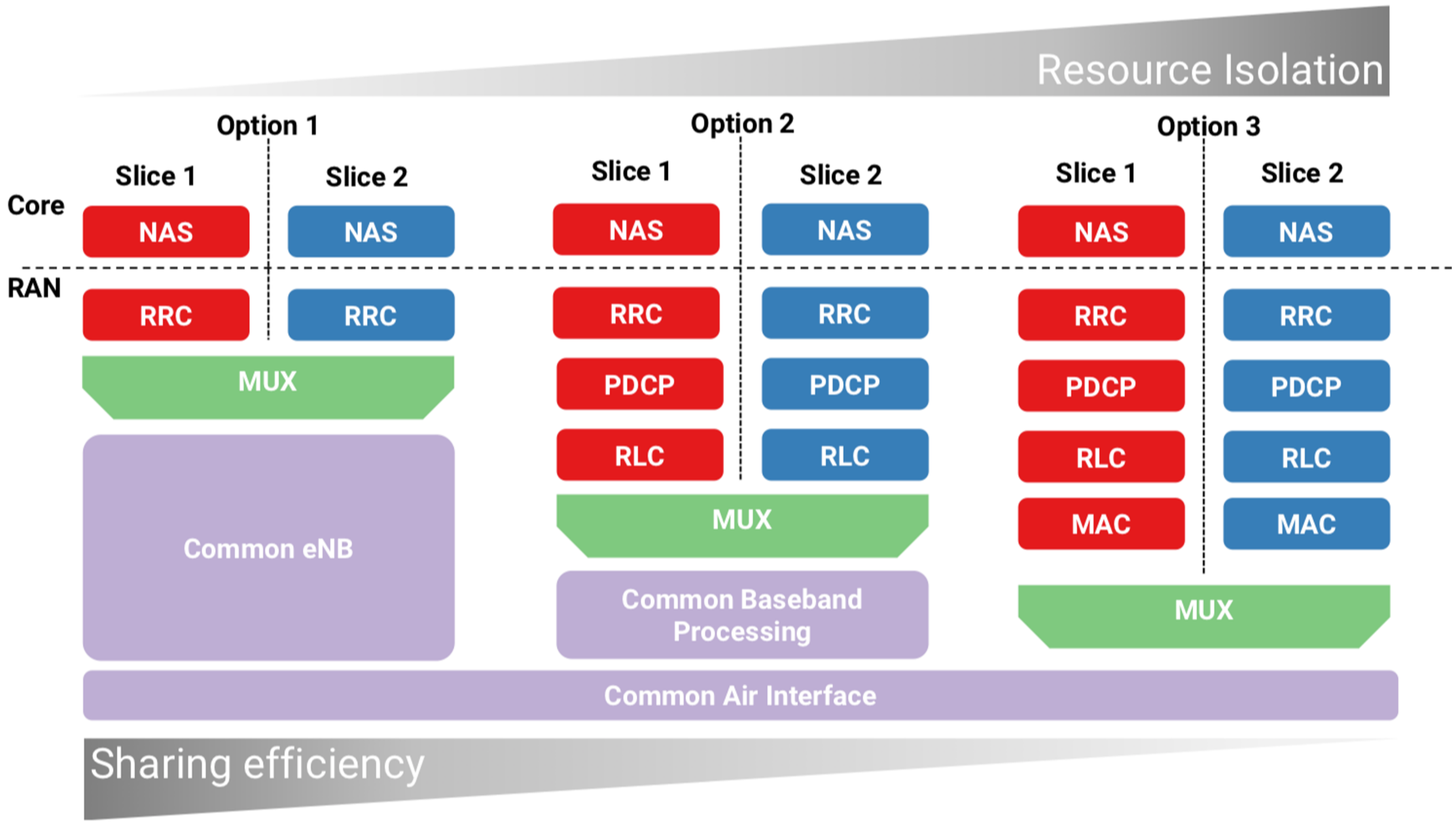
SRS vs OAI

- OAI (way) more CPU efficient



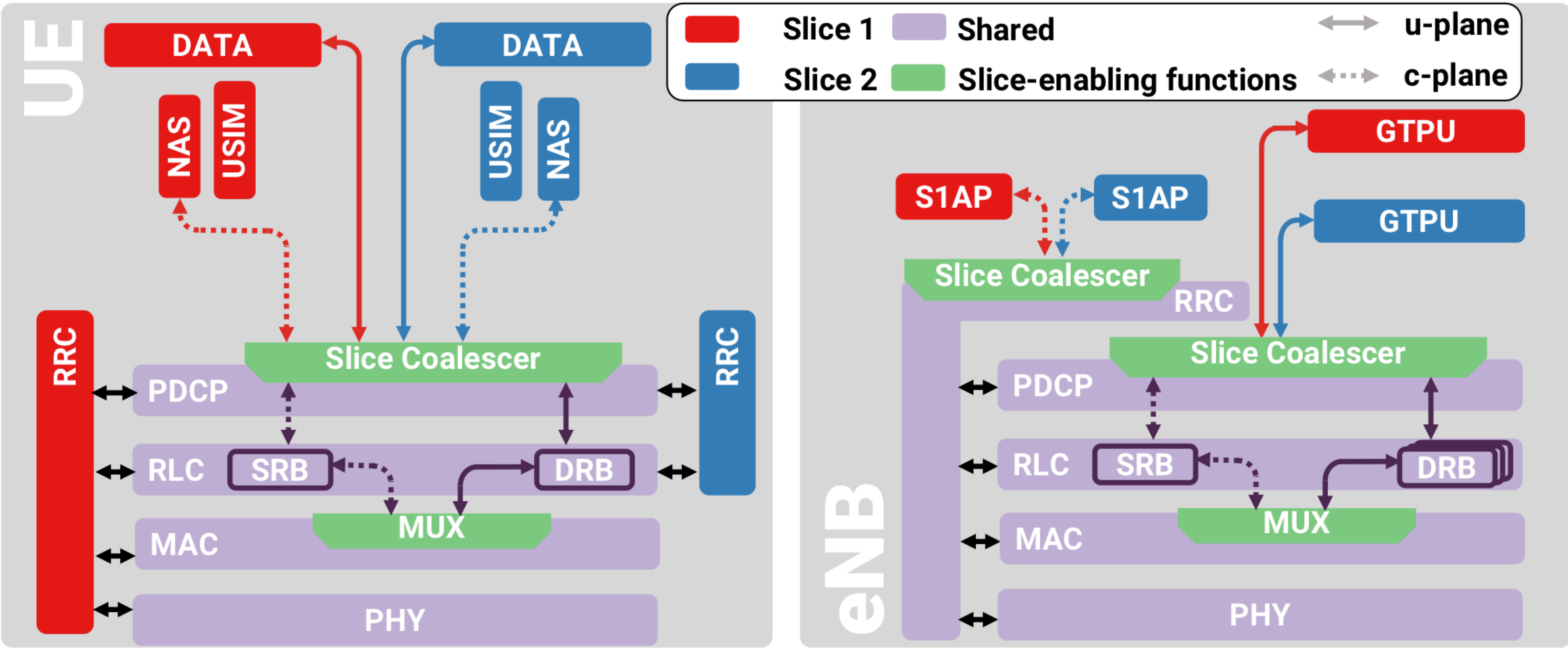
Slicing the RAN

- Traffic from different tenants has to be handled over the same spectrum



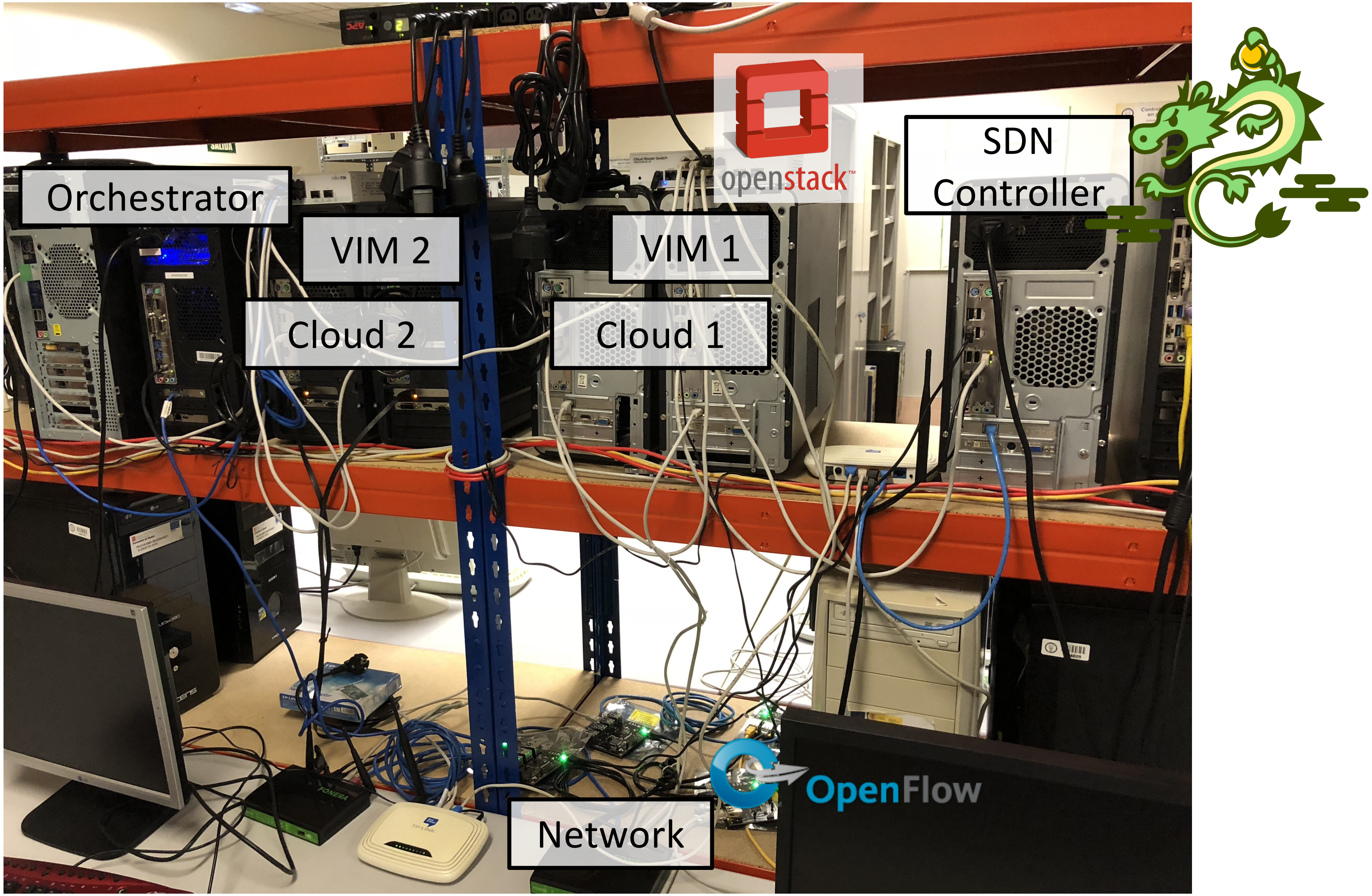
Adapted from: P. Rost et al., "Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks," in IEEE Comm. Mag.

Changes to the UE and eNB



https://github.com/wnluc3m

Core part



Orchestrator
VIM 2
Cloud 2
VIM 1
Cloud 1
openstack
SDN
Controller
OpenFlow
Network

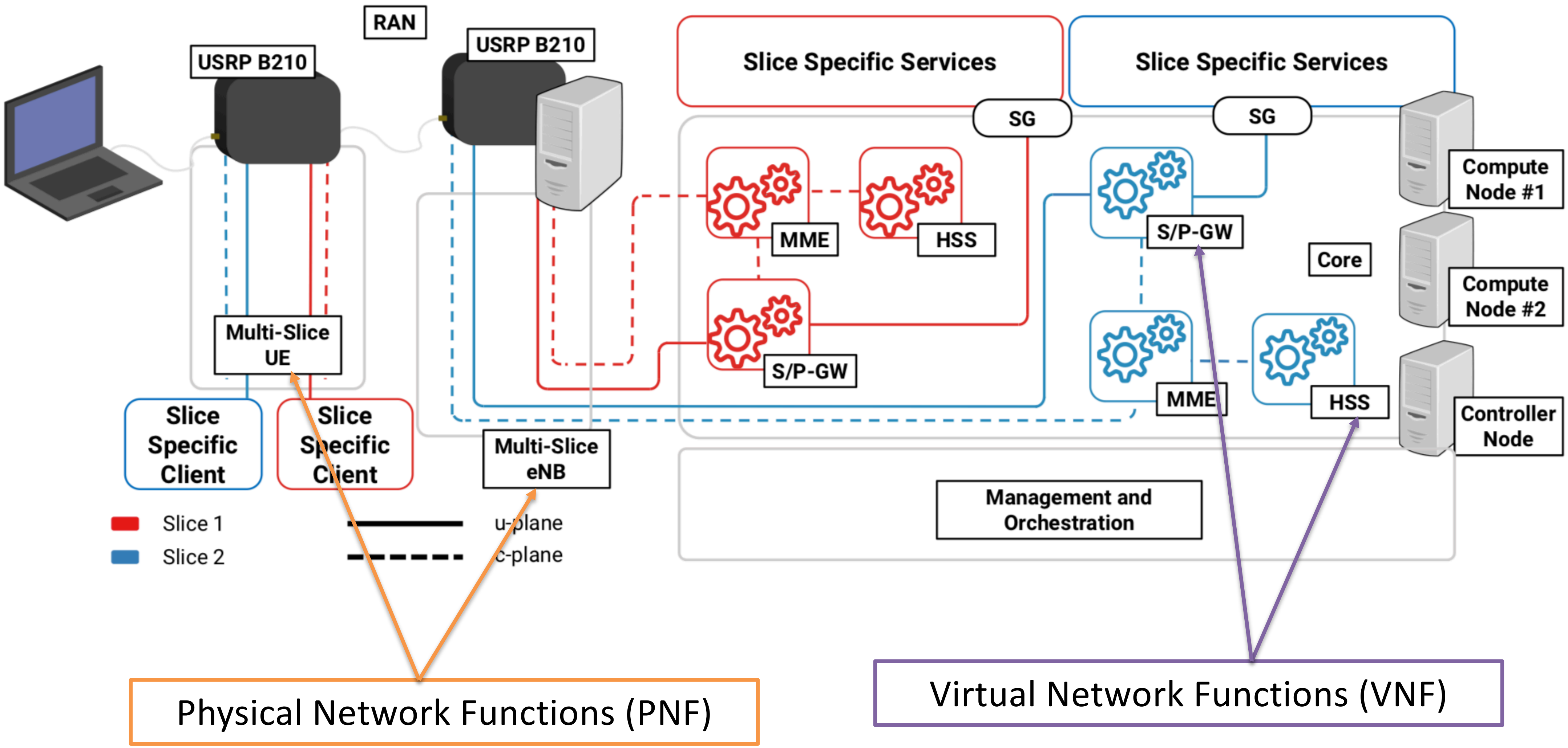
Non-radio part - Functions

- Core Network functions
  – Both OAI and SRS provide working software

- Management and orchestration
  – Ad hoc, through Command Line Interface (CLI)

Work	Base Software	Main purpose	Main Feature	Limitations	Open Source
Mendes et al. [5]	srsLTE	RAN Slicing	Multiple, per tenant, eNB virtualization .	Implementation only up to MAC layer.	No
Chang et al. [6]	OAI	RAN Slicing	Thorough evaluation of slices utilization	RAN Slicing only.	No
Foukas et al. [7]	OAI	RAN Slicing	SDN-based RAN slicing	It does not include a core.	Download upon request
Foukas et al. [8]	OAI	End-to-end slicing	Core Network handling multiple slices	Single Slice UE only.	No
POSENS	srsLTE	End-to-end slicing	Slice aware shared RAN.	One RAN split available.	Yes

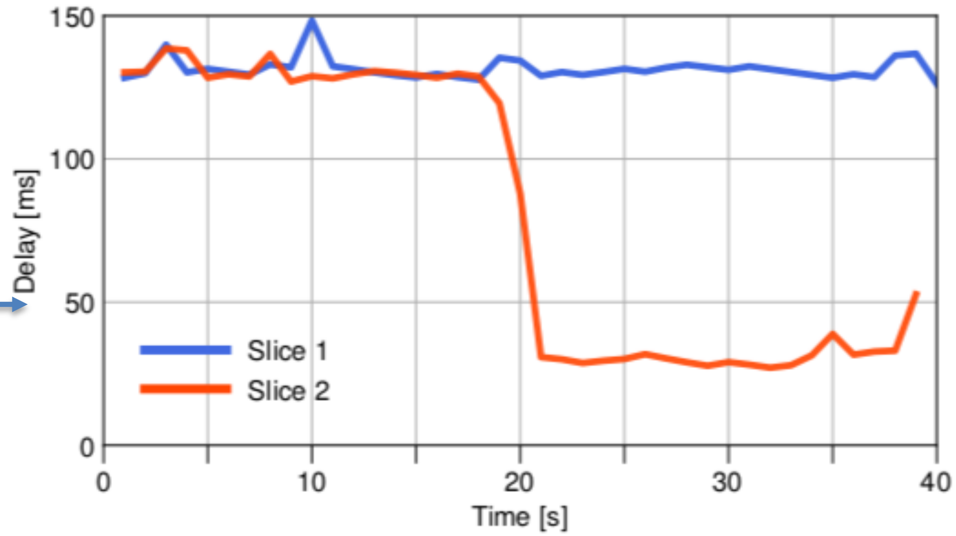
Overview

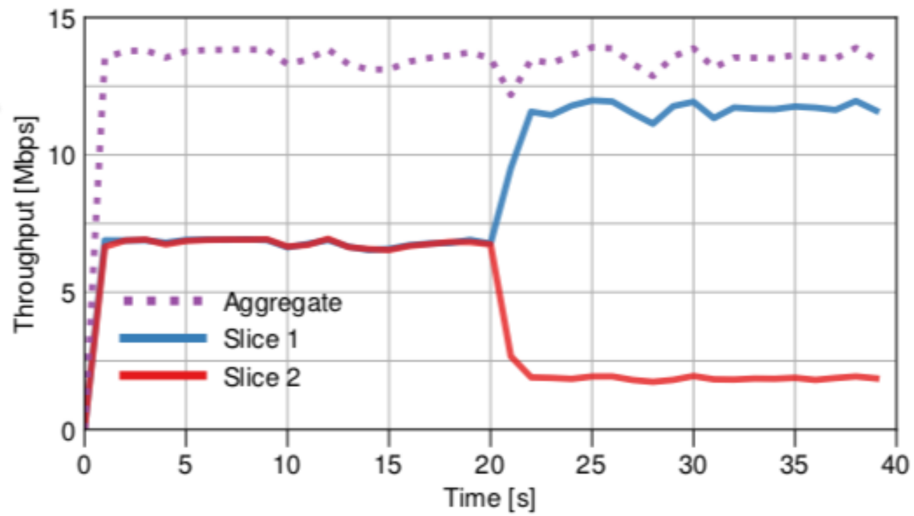


RAN
USRP B210
USRP B210
Multi-Slice UE
Slice Specific Client
Slice Specific Client
Multi-Slice eNB
Slice 1
Slice 2
u-plane
c-plane
Slice Specific Services
Slice Specific Services
SG
SG
MME
HSS
S/P-GW
S/P-GW
MME
HSS
Core
Compute Node #1
Compute Node #2
Controller Node
Management and Orchestration
Physical Network Functions (PNF)
Virtual Network Functions (VNF)

Results

- It works!
  - Function re-allocation
    - GW brought closer
  - Service re-composition
    - Added shaper + fw





- Also with a Nexus-5
  - Equipped with a programmable SIM card

SUMMARY & WRAP UP

Take away messages

- Orchestration strategies
  - Trade-off between isolation and efficiency
  - Static algorithms -> 2x resources even at the CN
  - Dynamic allocations: short timescales
  - Loose QoS requirements to boost efficiency

- Virtualization of Network Functions
  - Need to re-think their operation
  - Introduce resource awareness (PHY, CPU, load)

The future of Network Slicing

- Several efforts (more or less) aligned
  - Evolution of SDN, NFV
  - Adopted by 5G, IETF, etc.
  - Many ongoing initiatives (incl. Open source)
  - Trials (e.g., 5G-EVE, 5TONIC)
- Several challenges
  - Orchestration of these efforts
  - Security, Privacy, Resiliency
  - Use of Machine Learning
  - Practical experimentation

5G EVE

Thanks! Questions?

pablo@it.uc3m.es
@pablo_uc3m
http://www.it.uc3m.es/pablo/