

# vrAIIn

## A Deep Learning Approach to Virtualized Radio Access Networks (vRAN)

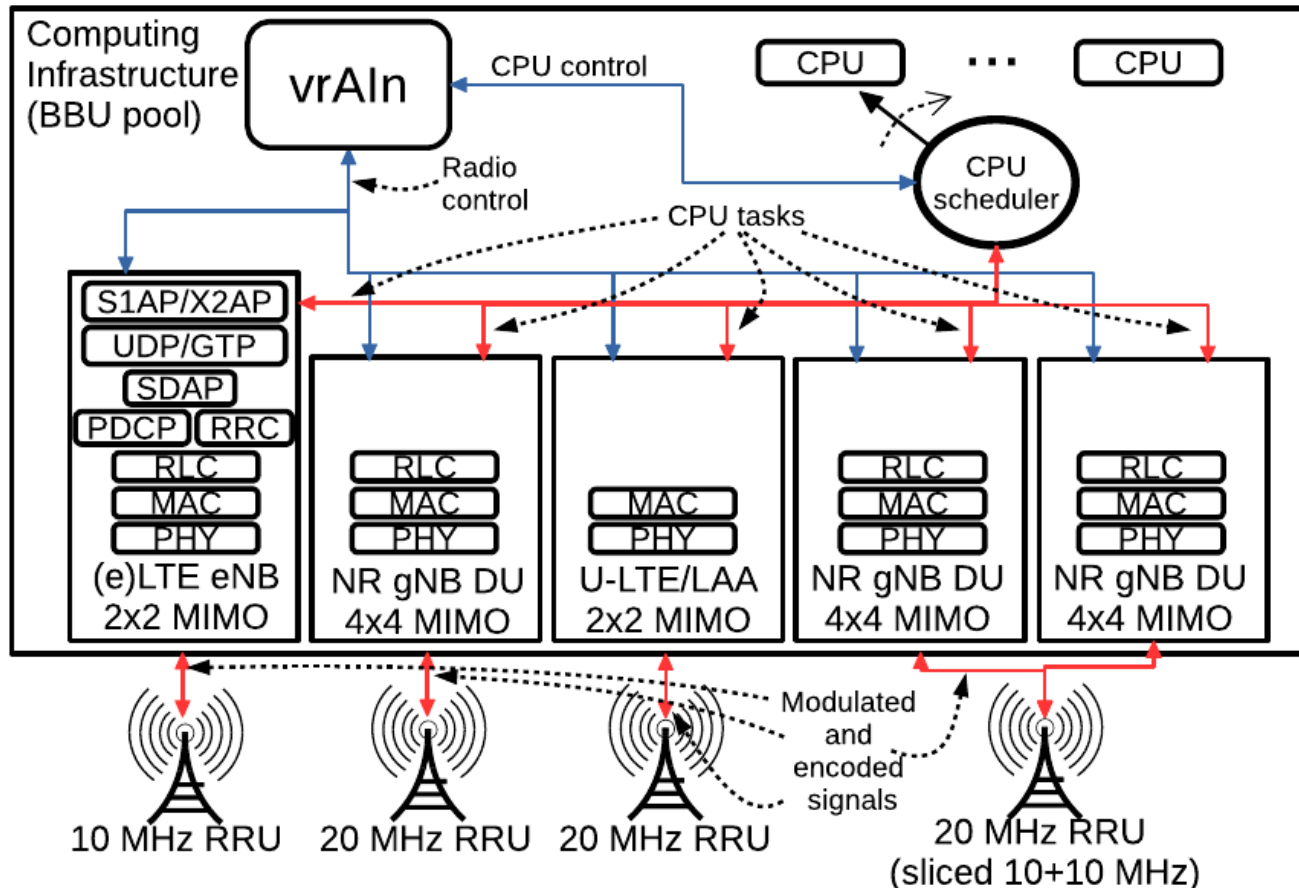
Andres Garcia-Saavedra

**NEC Laboratories Europe**

5GPPP, TB meeting, 27/05/2020

Virtualized RAN (vRAN) **centralizes software** radio access points (RAPs<sup>1</sup>) into **commodity** general-purpose computing infrastructure.

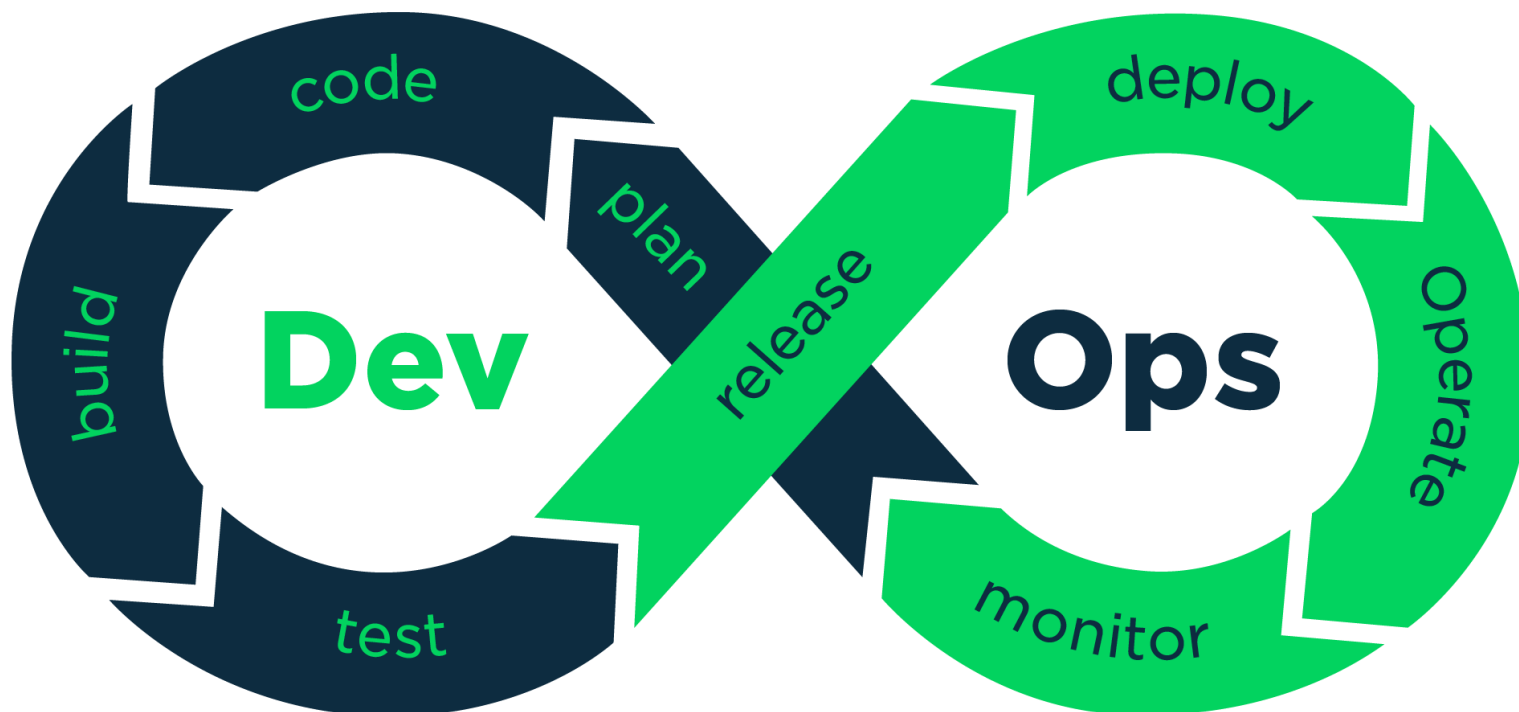
- **Advantage 1:** Statistical multiplexing gains from resource pooling (via **centralization**)



<sup>1</sup>Base Transceiver Station (BTS) in 2G, NodeB in 3G, enhanced NodeB (eNB) in 4G, next-generation NodeB (gNB) in 5G, etc.

Virtualized RAN (vRAN) **centralizes softwarized** radio access points (RAPs<sup>1</sup>) into **commodity** general-purpose computing infrastructure.

- **Advantage 1:** Statistical multiplexing gains from resource pooling (via **centralization**)
- **Advantage 2:** Agile update roll-ups such as security patches, protocol upgrades, bug fixes, CI/CD, agile, DevOps, etc. (via **softwarization**)

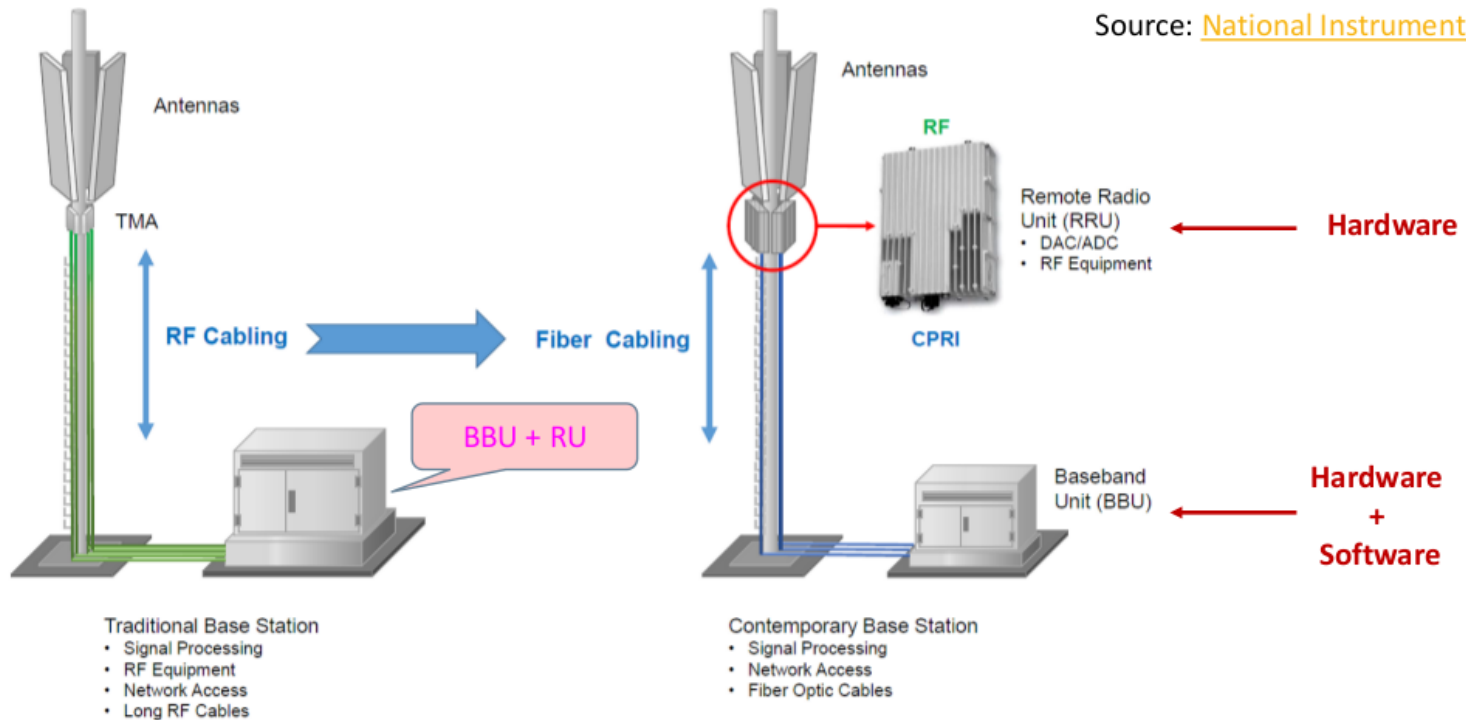


<sup>1</sup>Base Transceiver Station (BTS) in 2G, NodeB in 3G, enhanced NodeB (eNB) in 4G, next-generation NodeB (gNB) in 5G, etc.

Virtualized RAN (vRAN) **centralizes softwarized** radio access points (RAPs<sup>1</sup>) into **commodity** general-purpose computing infrastructure.

- **Advantage 1:** Statistical multiplexing gains from resource pooling (via **centralization**)
- **Advantage 2:** Agile update roll-ups such as security patches, protocol upgrades, bug fixes, CI/CD, agile, DevOps, etc. (via **softwarization**)
- **Advantage 3:** Cheaper deployment and new business opportunities (via **commoditization**)

Contemporary RAN



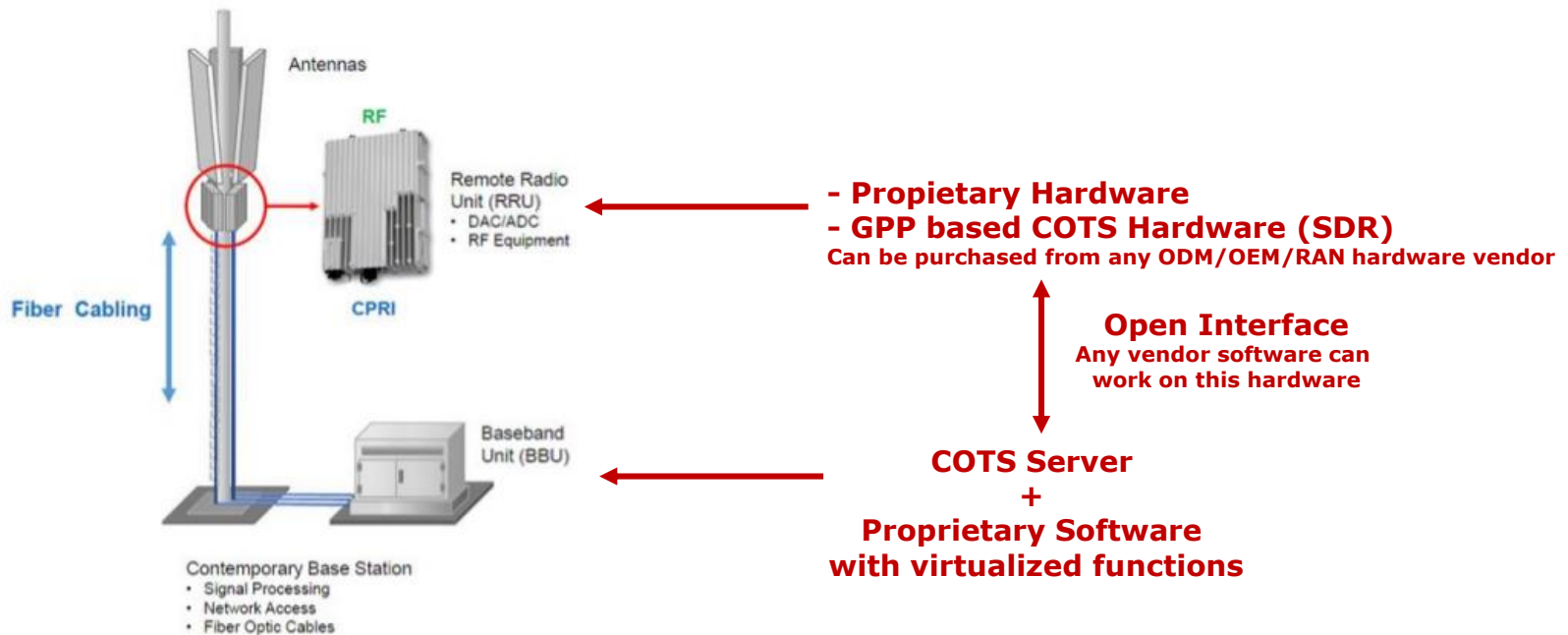
\*From 3G4G Blog (<http://www.3g4g.co.uk/>)

<sup>1</sup>Base Transceiver Station (BTS) in 2G, NodeB in 3G, enhanced NodeB (eNB) in 4G, next-generation NodeB (gNB) in 5G, etc.

Virtualized RAN (vRAN) **centralizes softwarized** radio access points (RAPs<sup>1</sup>) into **commodity** general-purpose computing infrastructure.

- **Advantage 1:** Statistical multiplexing gains from resource pooling (via **centralization**)
- **Advantage 2:** Agile update roll-ups such as security patches, protocol upgrades, bug fixes, CI/CD, agile, DevOps, etc. (via **softwarization**)
- **Advantage 3:** Cheaper deployment and new business opportunities (via **commoditization**)

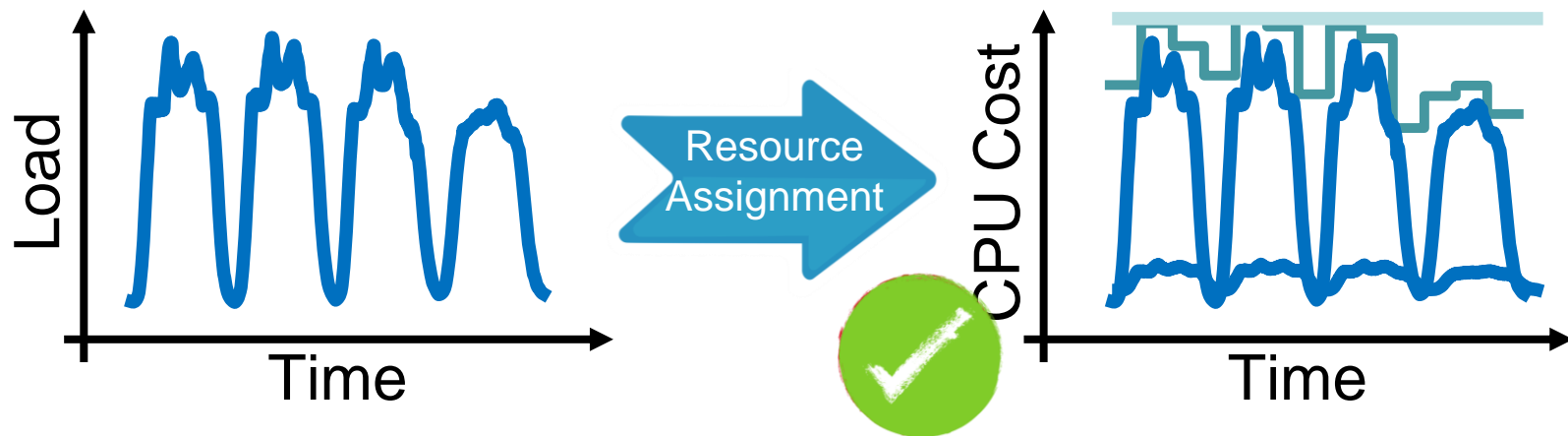
vRAN  
(Open RAN)



\*From 3G4G Blog (<http://www.3g4g.co.uk/>)

<sup>1</sup>Base Transceiver Station (BTS) in 2G, NodeB in 3G, enhanced NodeB (eNB) in 4G, next-generation NodeB (gNB) in 5G, etc.

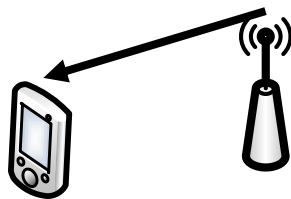
# The resource orchestration problem



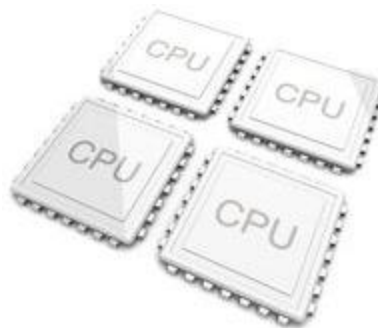
Resource assignment depends on many factors such as...



User Demand



Channel Conditions

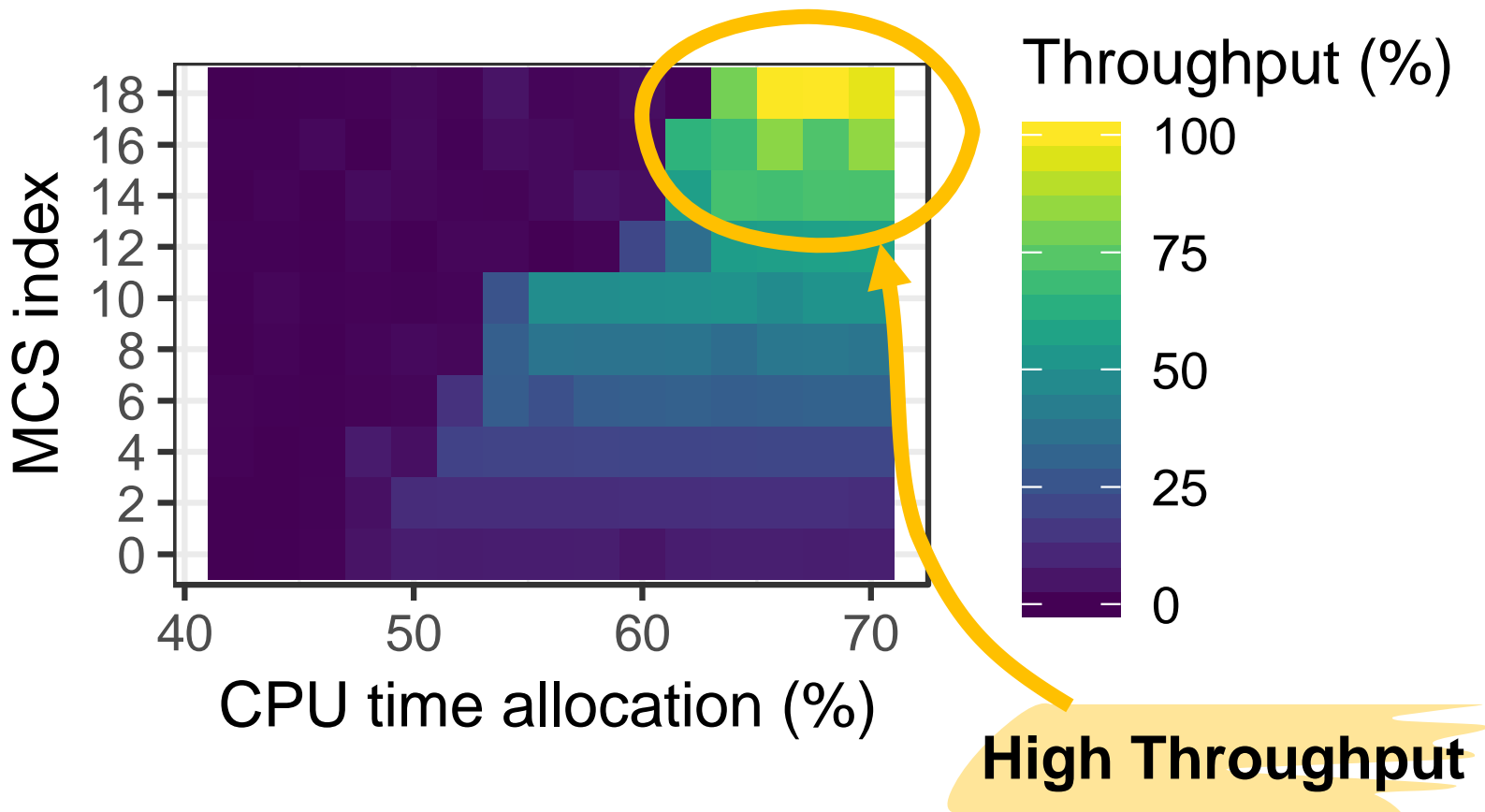


CPU Platform

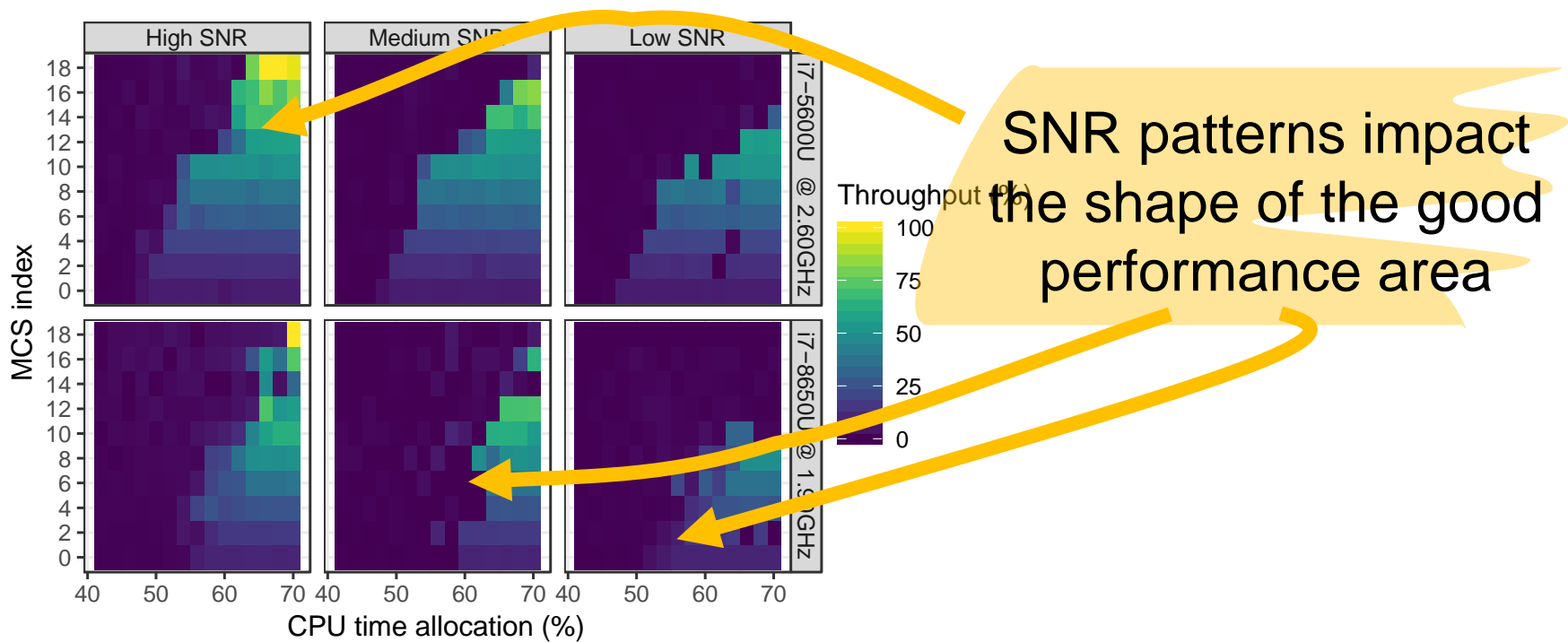


SW stack, functional split

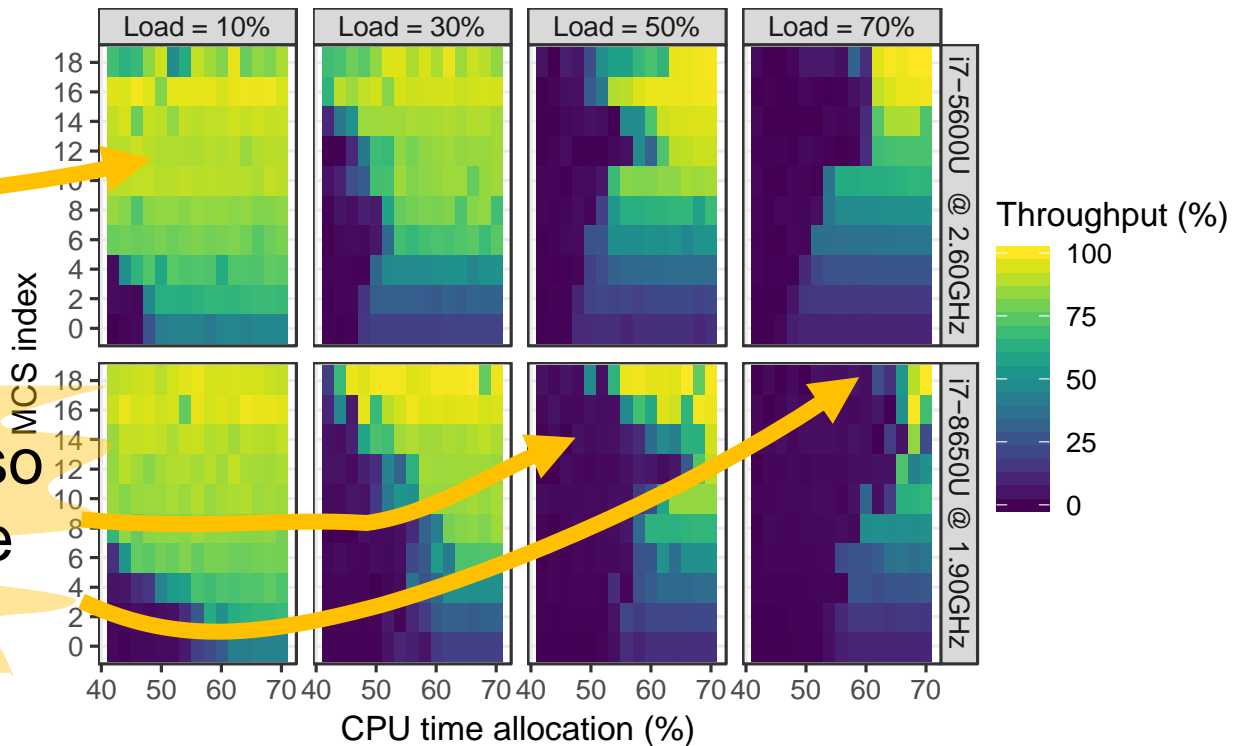
# The resource orchestration problem



# The problem is far from trivial



# The problem is far from trivial



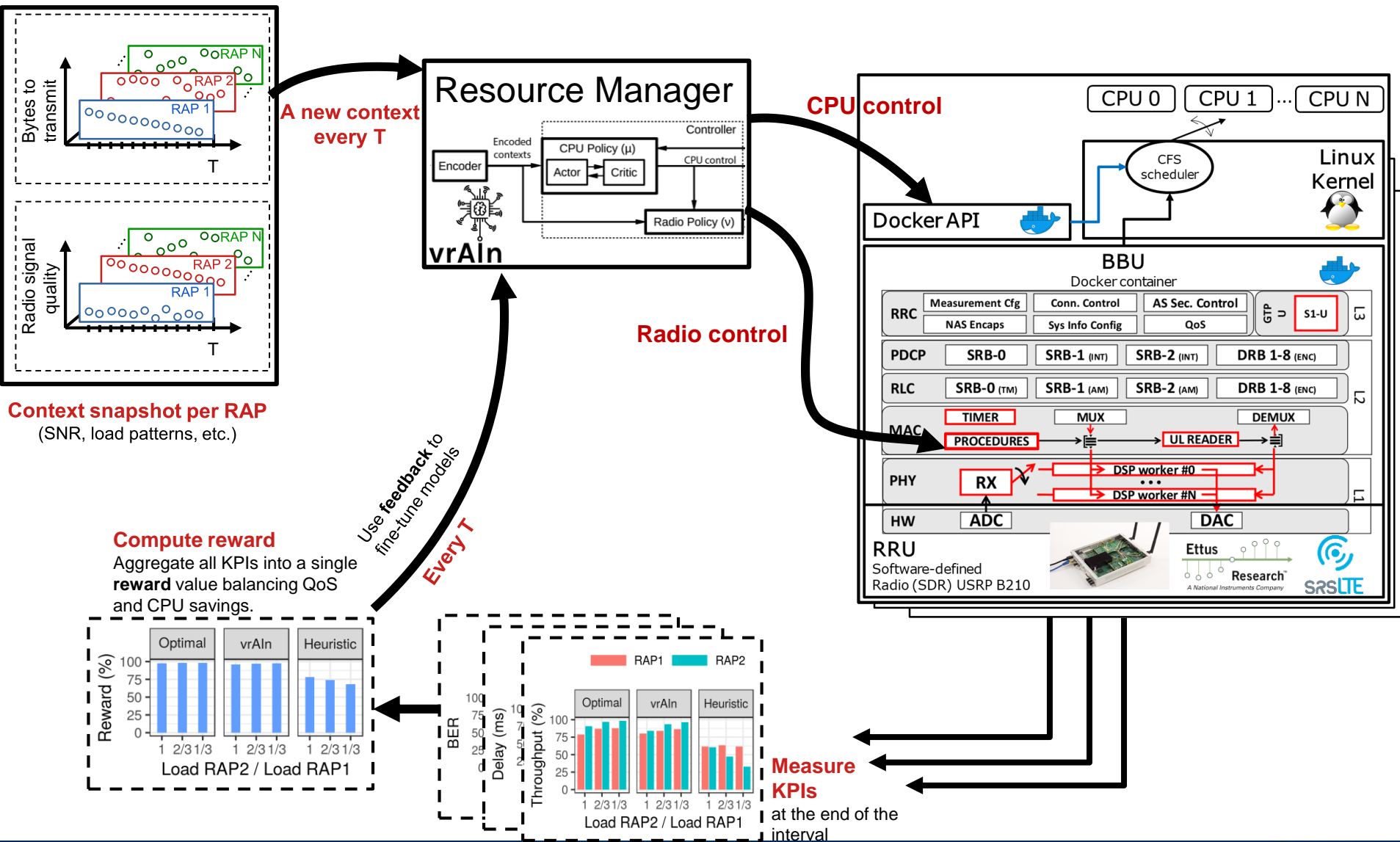
Performance also depends on the traffic patterns

# The problem is far from trivial

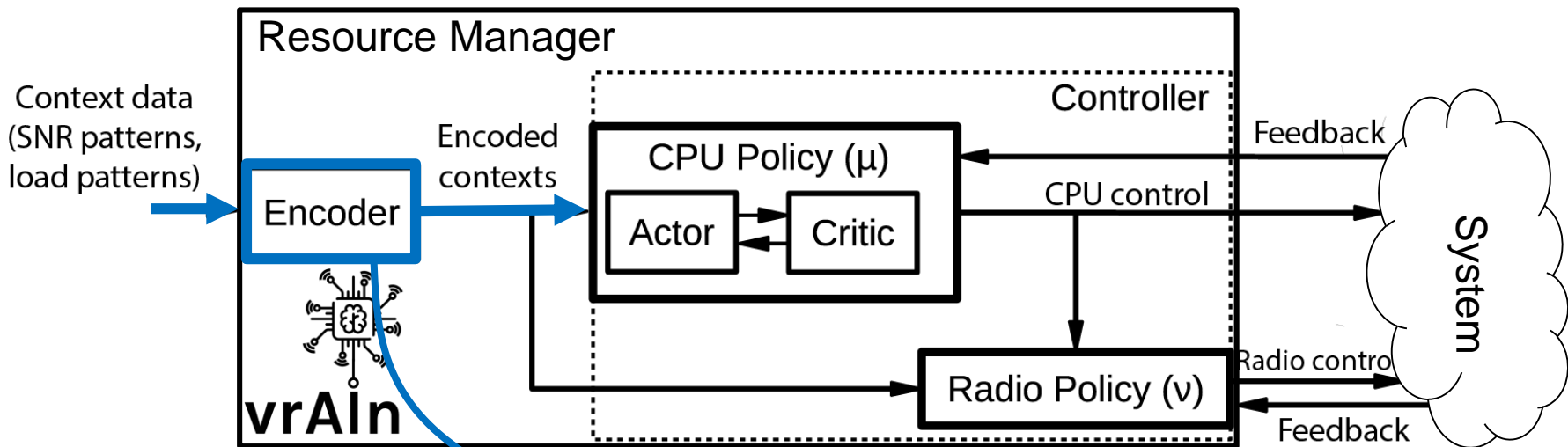
---

Performance is a very complex function of  
the contexts and the resource assignment  
→ Deep Learning

# vrAIn: AI based vRAN resource controller

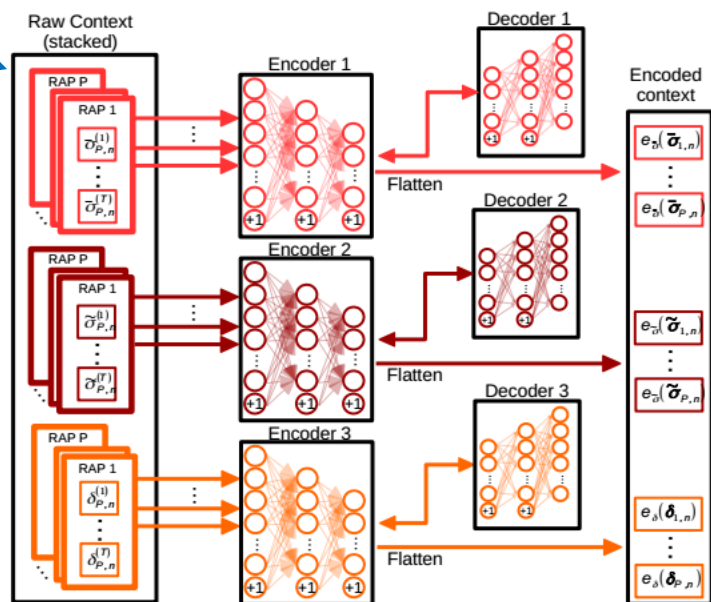


# vrAIn: Challenges and Solutions

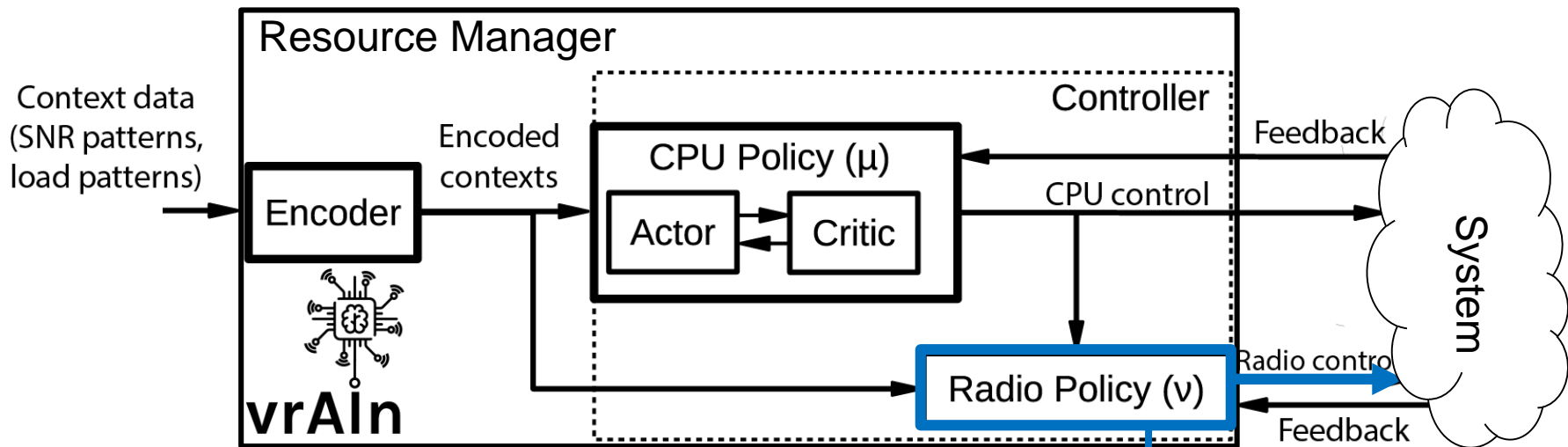


**Challenge #1:**  
Dimensionality of the input contexts

**Solution:**  
3 Sparse autoencoders to reduce the dimensionality



# vrAIn: Challenges and Solutions

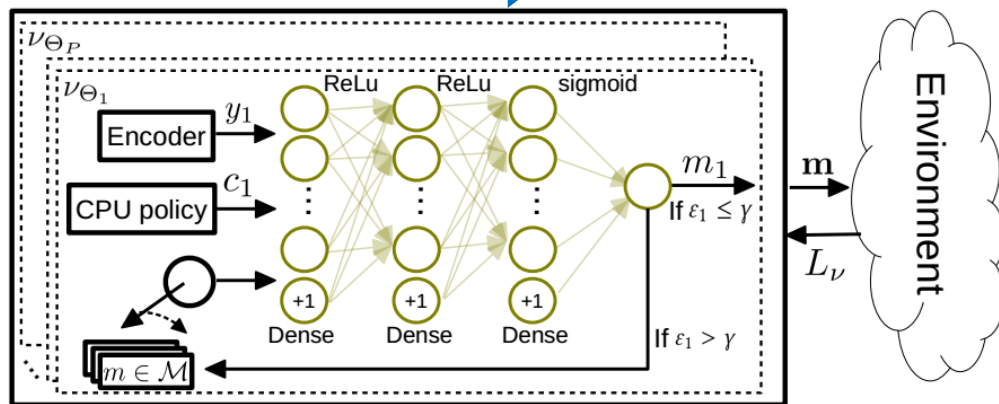


## Challenge #2:

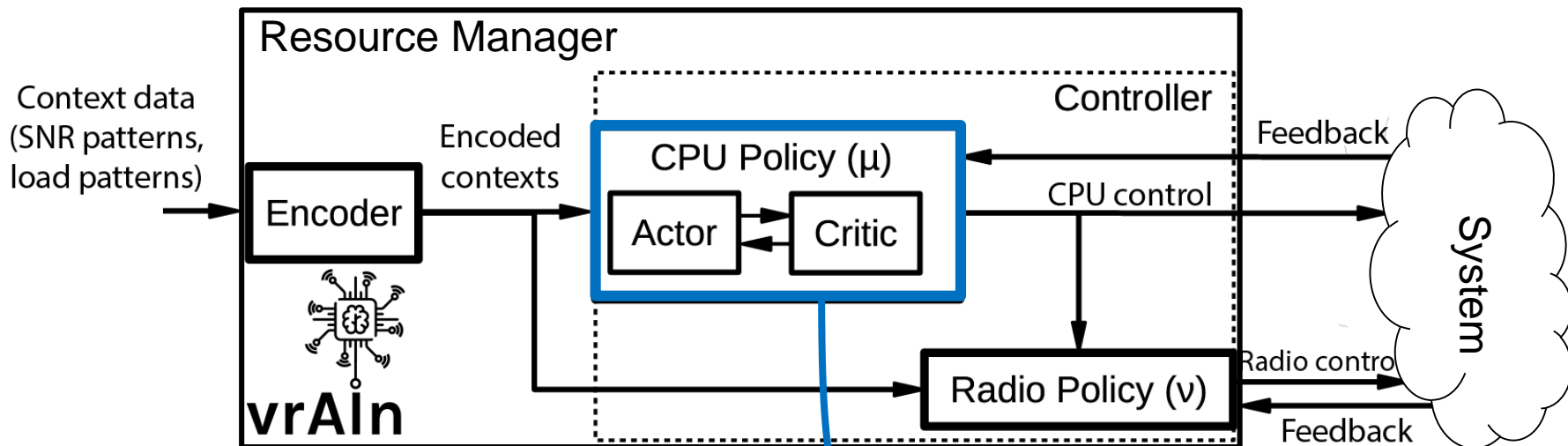
Heterogeneity of the action space (continuous and discrete)

## Solution:

Decoupling of the radio and the CPU policy

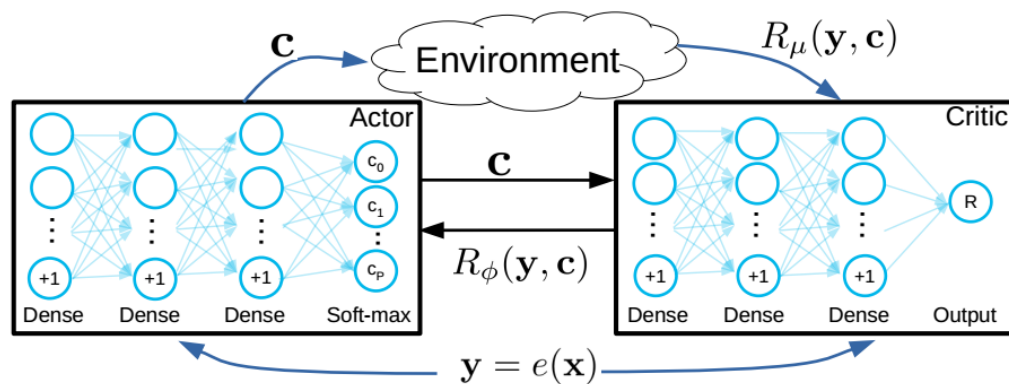


# vrAIn: Challenges and Solutions



**Challenge #3:**  
N-dimensional continuous controls for the CPU policy

**Solution:**  
Deep Deterministic Policy Gradient



# Evaluation results: Unlimited Resources

---

## Scenario 1

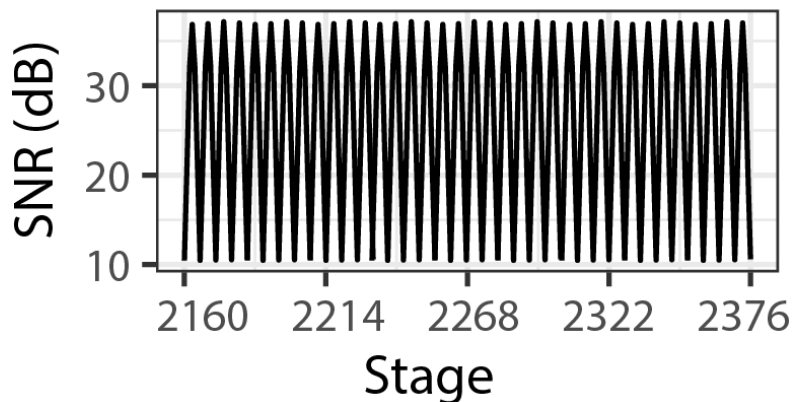
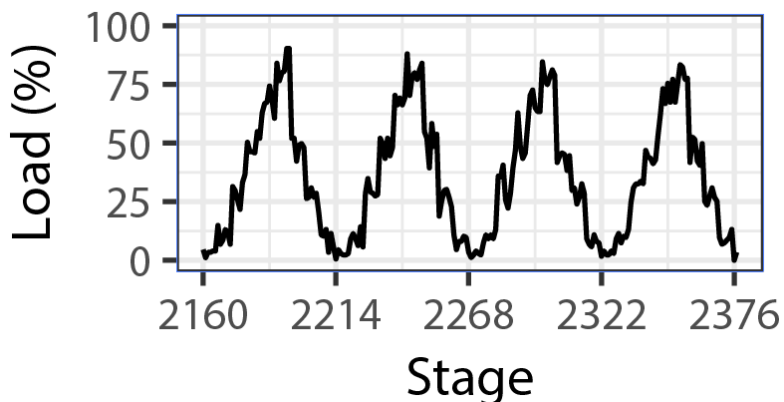
- Unlimited CPU resources
- One virtual Base Station

### Objective:

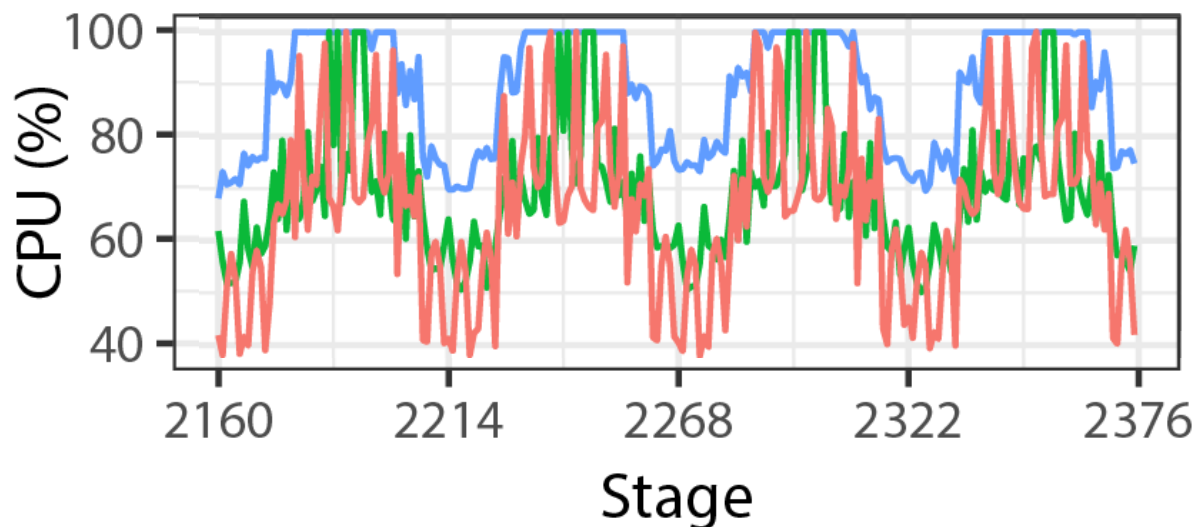
- Minimize the costs while satisfying the QoS

# Evaluation results: Unlimited Resources

Contexts:



vrAIn CPU allocation:



Average CPU Savings

High QoS	14%
Medium QoS	26%
Low QoS	39%

## Scenario 2

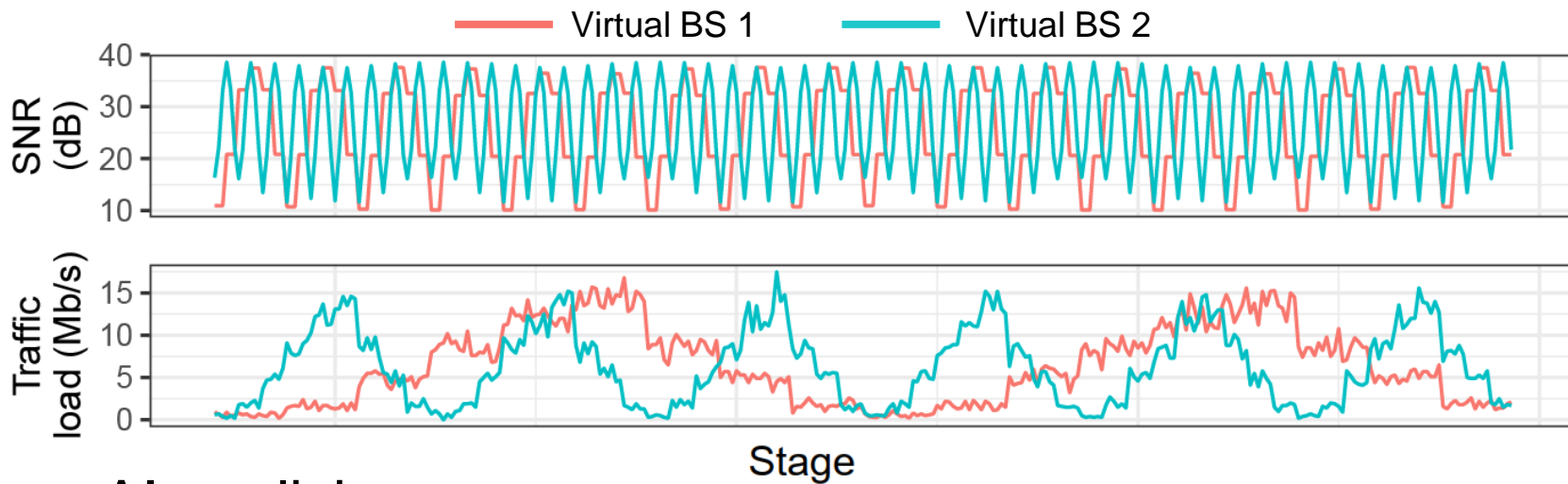
- Limited CPU resources (one core)
- Two virtual Base Station

### Objective:

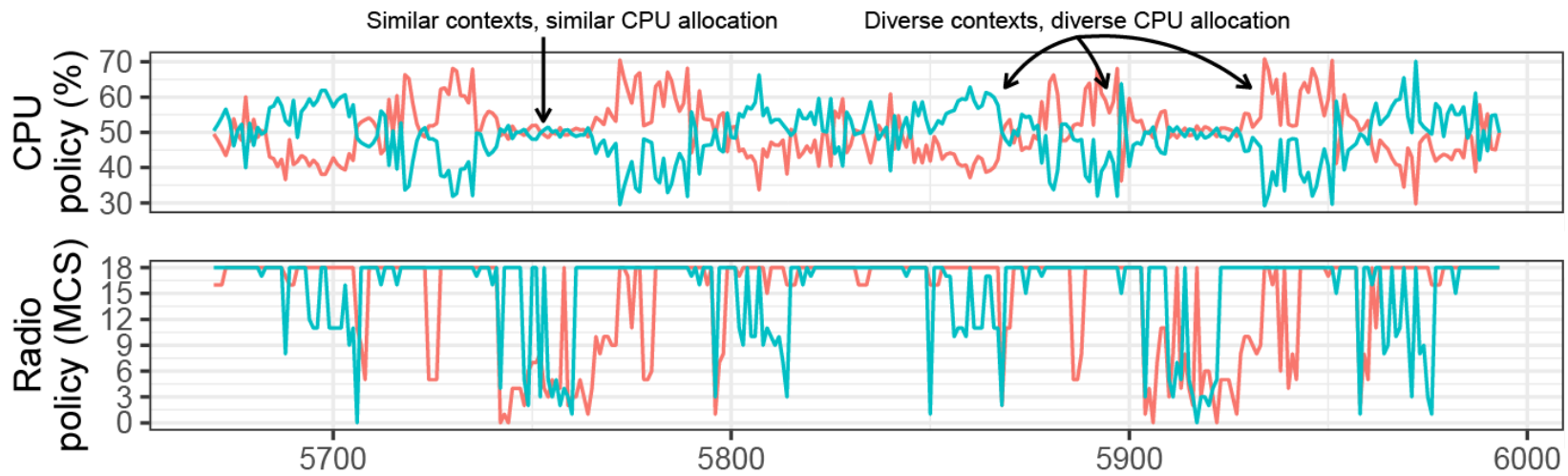
- Maximize the performance of both virtual BSs

# Evaluation results: Limited Resources

## Contexts:



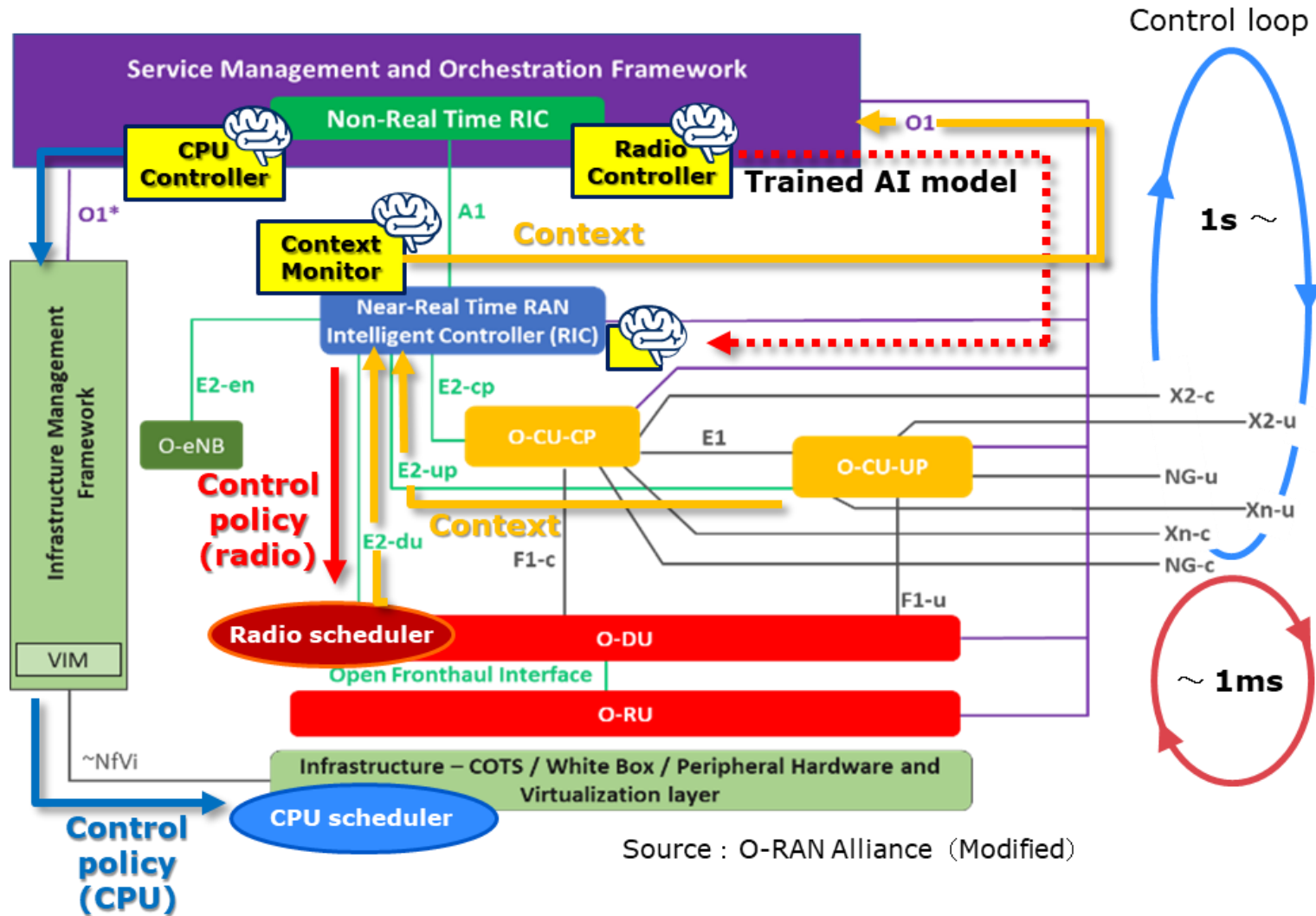
## vrAI policies:



vrAI achieves zero decoding error

- The performance of a virtual BS is a very complex function of the contexts and the resource assignment, motivating the use of **Deep Learning**.
- We solve the problem using a novel combination of **Sparse Autoencoders**, a **Reinforcement Learning** algorithm and a **Neural Network Classifier**.
- Our solution **minimizes the costs** with unlimited resources and **maximizes the performance** with limited resources. With respect to state-of-the-art solutions, vrAI achieves...
  - **CPU savings ~30% with unlimited resources.**
  - **Throughput increase ~25% per virtual Base Station.**
- We trained our models with **real data** and implemented a **proof-of-concept** of the solution.
  - **Dataset in <https://github.com/agsaaved/vrain>**

# Integration of vrAIIn into O-RAN



## Decoding Error Probability

Empirically computed by sampling every subframe (UL) or via HARQ

## CPU allocation

$$r(\mathbf{x}, \mathbf{a}) := \sum_{i \in \mathcal{P}} \mathbb{P} [q_{i, x_i, a_i} < Q_i] - M \epsilon_i - \lambda c_i$$

## Buffer State (random variable)

Empirically estimated by sampling every Buffer State Report (BSR) in UL

## Target mean buffer state (bytes)

An easy measure of delay

## Constant Parameters

M is a large value  
λ is a small value