DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
800 ANNI 1222·2022
UNIVERSITÀ DEGLI STUDI DI PADOVA
SIGNET
signet.dei.unipd.it

Communication and Learning in Dynamic Networks

Federico Mason

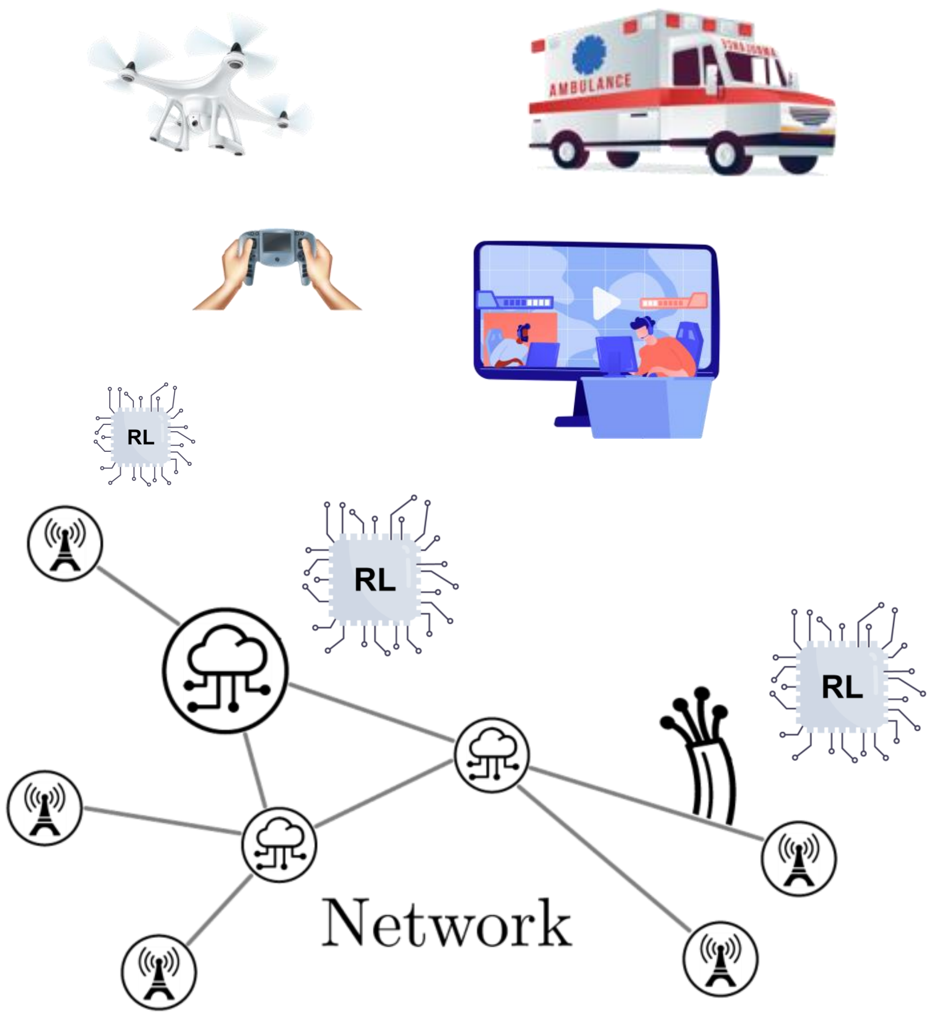
Supervisor: Prof. Andrea Zanella

Email: masonfed@dei.unipd.it

GTTI annual meeting, September 11, 2023

Introduction

- Modern networks are characterized by **heterogeneous** applications
- The **same architecture** must adapt to different scenarios in real time
- Network control is distributed among **multiple autonomous units**



Introduction

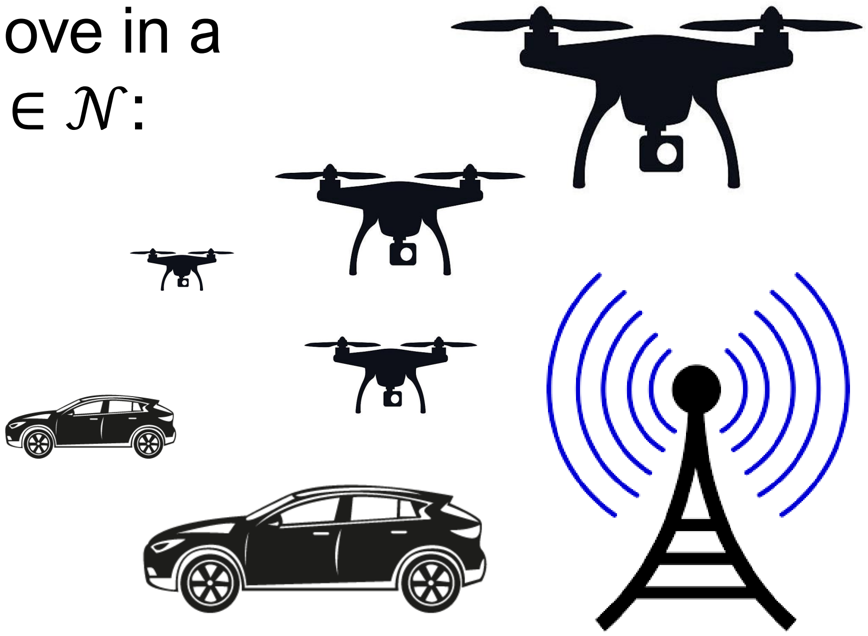
1. We need to estimate the overall network status
   → **Network Mapping**

2. We need to coordinate multiple control units
   → **Distributed Learning**

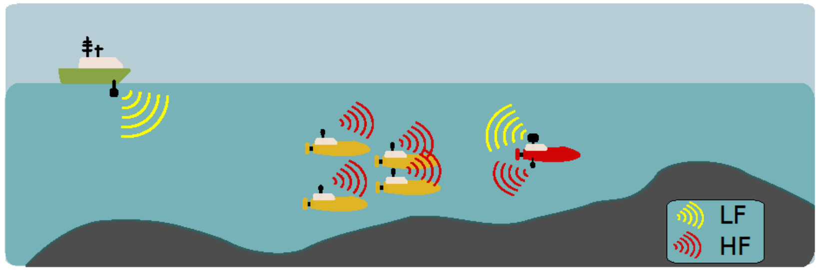
3. We need to balance resources among services
   → **Resource Allocation**

Network Mapping: Scenario

We consider a group of **autonomous nodes** that move in a 2D or 3D scenario. At each timeslot t, each node $n \in \mathcal{N}$:



- Uses local sensors to **observe** the environment and estimate its **own state**
- Spreads its state estimate according to some **communication** strategy
- Exploits the received information to **estimate** the overall **network state**



Network Mapping: Model

Each node $n \in \mathcal{N}$ is equipped with $|\mathcal{N}|$ **Unscented Kalman Filters (UKFs):**

- The node n uses one UKF to compute the state estimate $\hat{s}_{n,n}(t)$
- It uses the other UKFs to compute the state estimates $\hat{s}_{n,k}(t), \forall\, k \in \mathcal{N}:\ k \neq n$

Tracking model accuracy

Communication strategy

The **system performance** is given by:

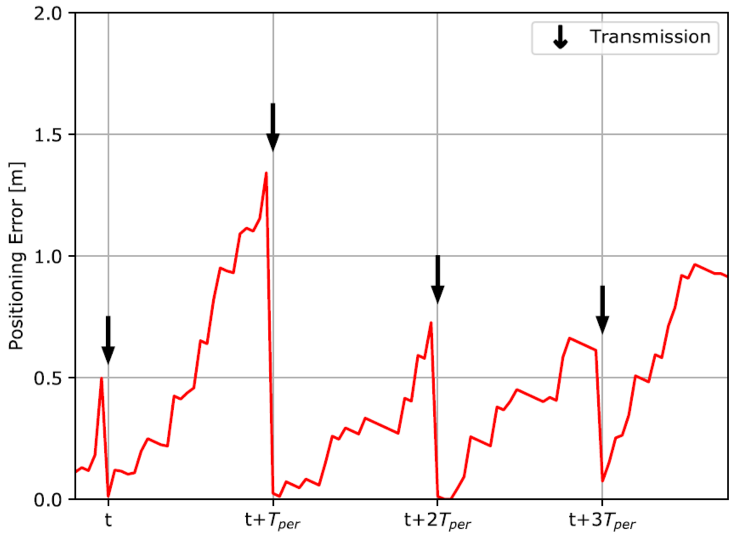
$$\Omega = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \left(\left| \hat{s}_{n,n}(t) - s_n(t) \right| + \sum_{k \in \mathcal{N}, k \neq n} \left| \hat{s}_{n,k}(t) - s_k(t) \right| \right)$$

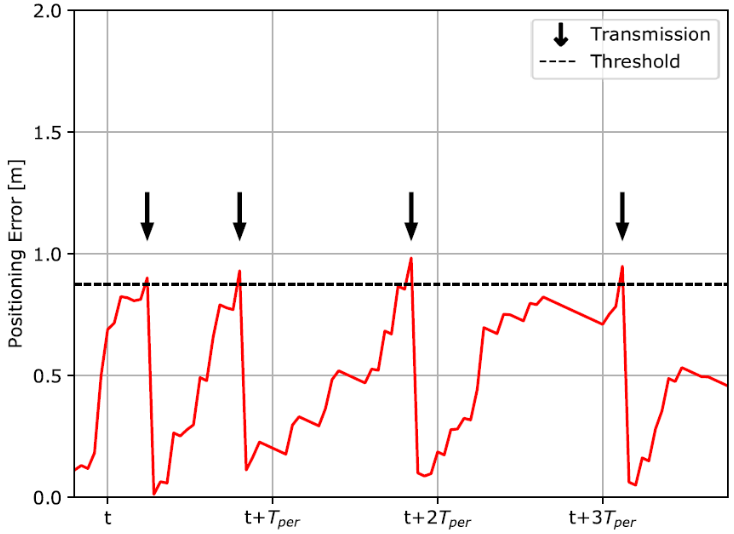
Network Mapping: Approach

Benchmark:
new communications are started according to the age of the information

New proposal:
new communications are started according to the urgency of the information

- Each node $n \in \mathcal{N}$ implements as additional UKF emulating the tracking process of the other nodes

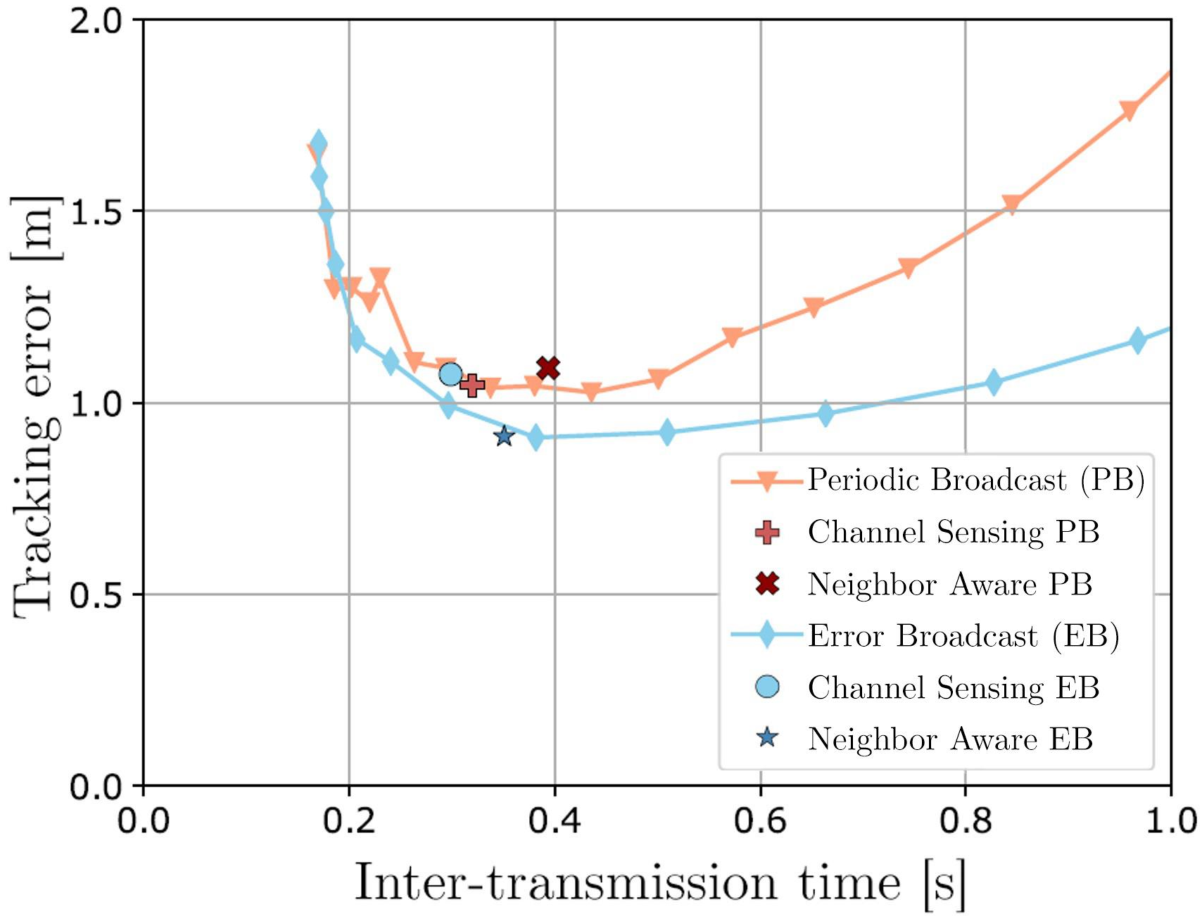




Network Mapping: Results

We tune communication according to:

- Channel sensing
- State estimation

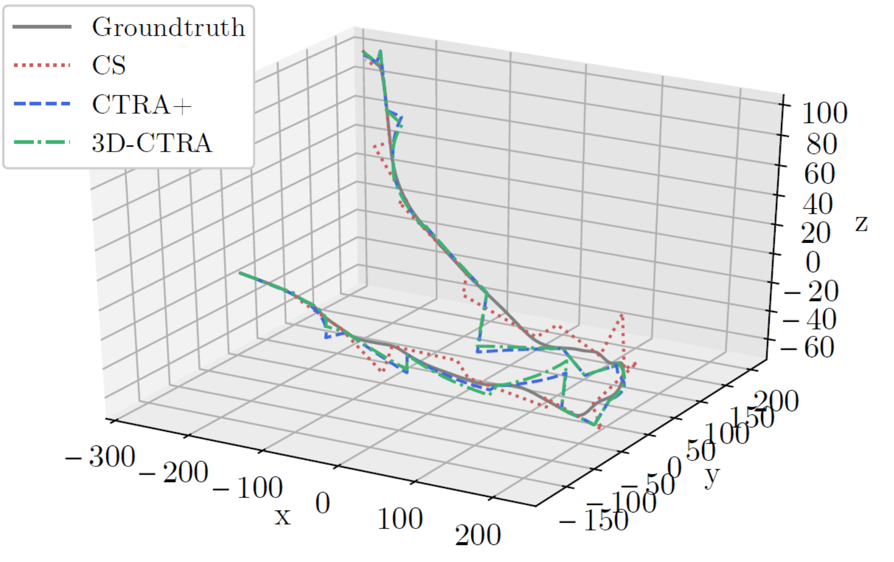


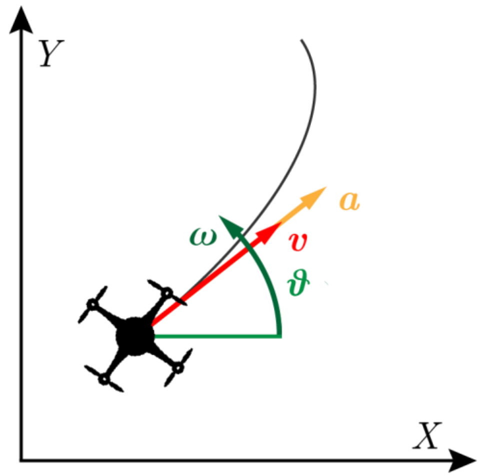
F. Mason, M. Giordani, F. Chiariotti, A. Zanella and M. Zorzi, "An Adaptive Broadcasting Strategy for Efficient Dynamic Mapping in Vehicular Networks," in *IEEE Transactions on Wireless Communications*, vol. 19, no. 8, pp. 5605-5620, May 2020

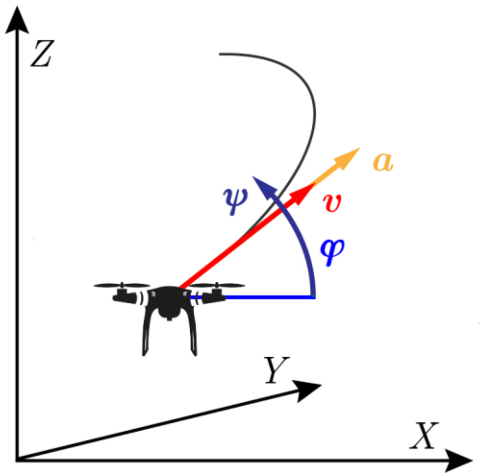
Network Mapping: Approach

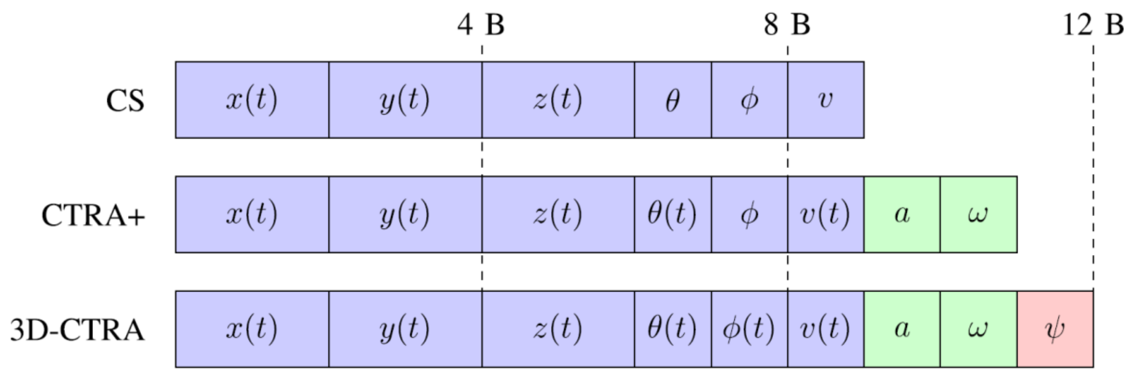
We design two new models, named **CTRA+** and **3D-CTRA**, to operate in 3D scenarios

- More advanced models lead to a higher **communication overhead**







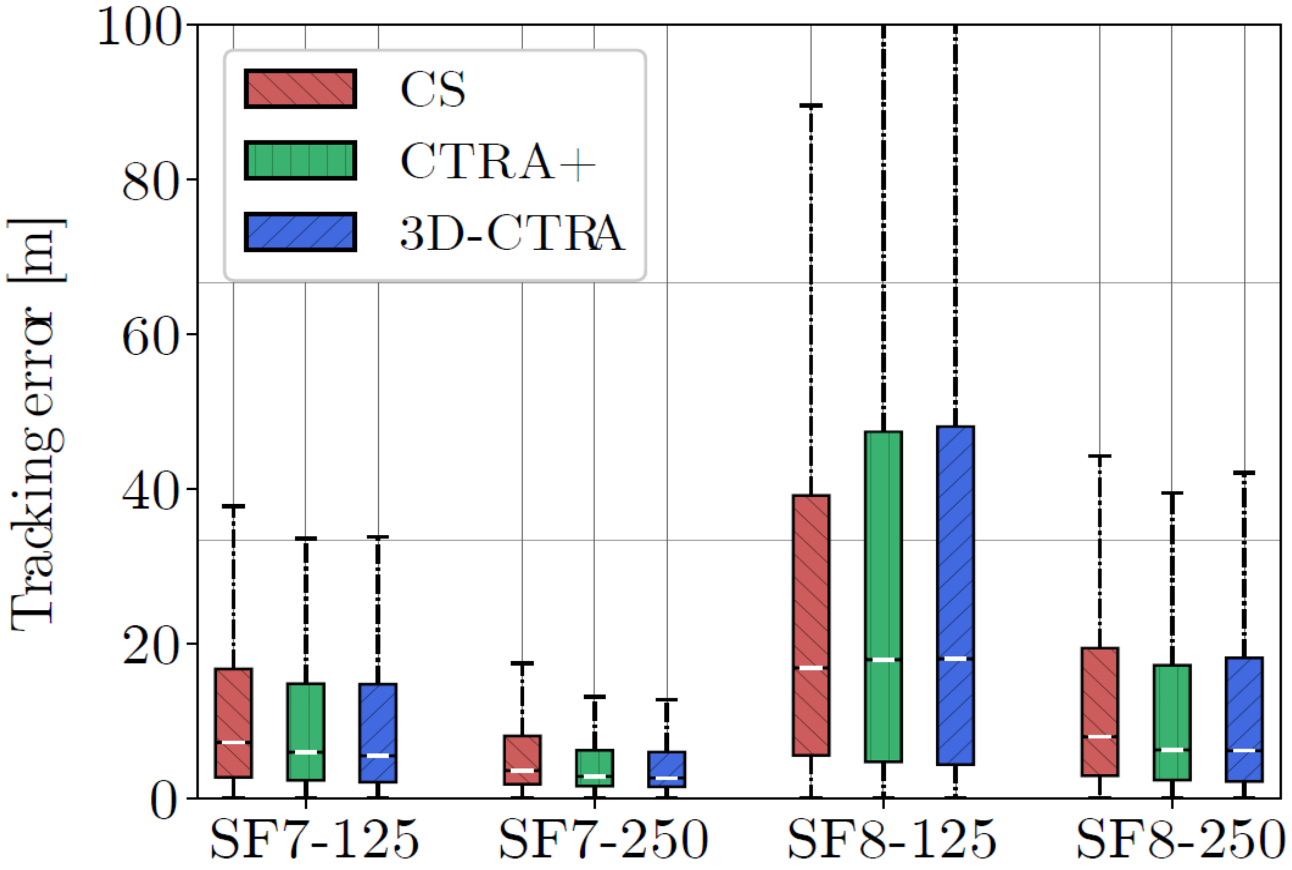


Network Mapping: Results

SF → Spreading Factor

CS → Constant Speed

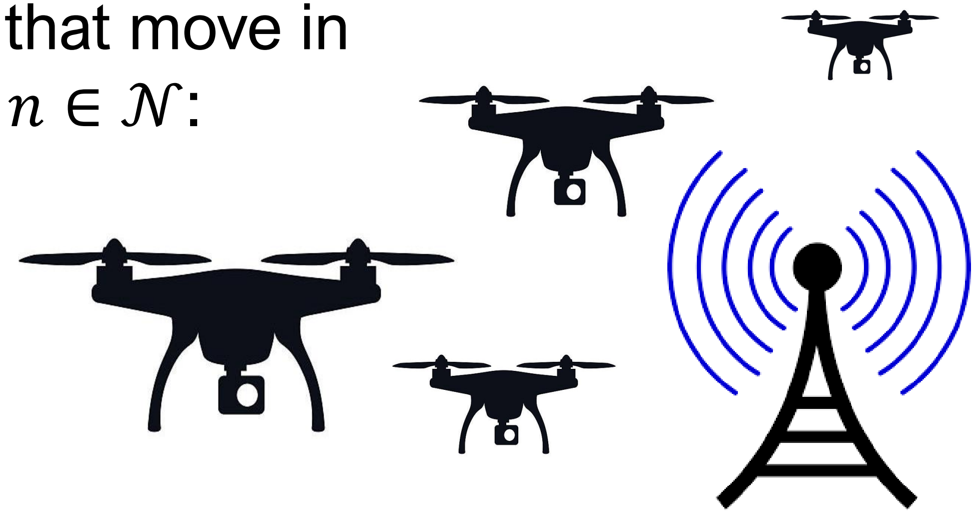
CTRA → Constant Turn Rate and Acceleration



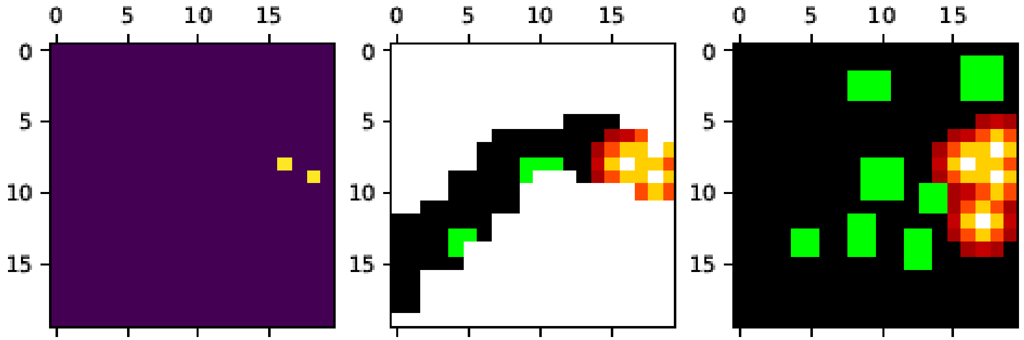
F. Mason, M. Capuzzo, D. Magrin, F. Chiariotti, A. Zanella and M. Zorzi, “Remote Tracking of UAV swarms via 3D mobility models and LoRaWAN communications," in *IEEE Transactions on Wireless Communications*, vol. 21, no. 5, pp. 2953-2968, October 2021

Distributed Learning: Scenario

We consider a group of **autonomous nodes** that move in a 2D scenario. At each timeslot t, each node $n \in \mathcal{N}$:

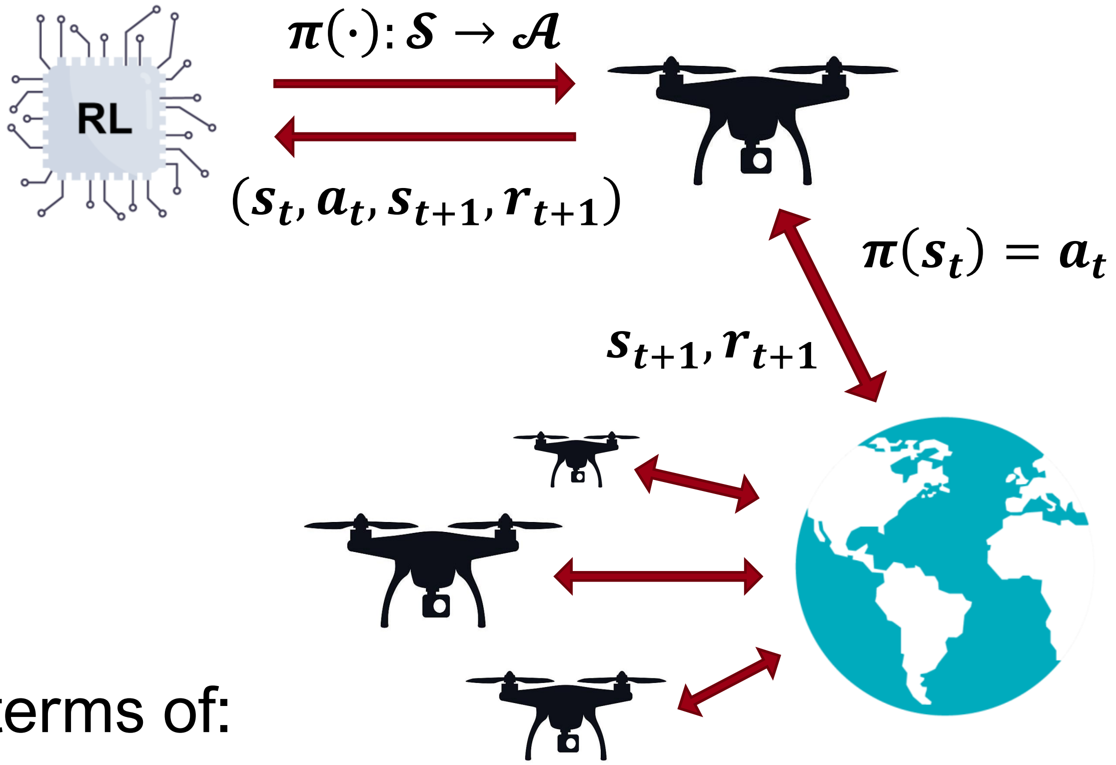


- Spreads local information according to some **communication** strategy
- Exploits the received information to **estimate** the overall network state
- Decides in which directions to **move** in order to reach a **target** location



Distributed Learning: Model

- Each node is controlled by a **Reinforcement Learning (RL)** agent computing its **policy** $\pi(\cdot): \mathcal{S} \rightarrow \mathcal{A}$
- At each **step** t, the agent observes the **state** $s_t \in \mathcal{S}$, takes an **action** $a_t \in \mathcal{A}$ and receives a **reward** r_{t+1}
- In our system, **multiple agents** interact with the **same environment**



Performance is given in terms of:

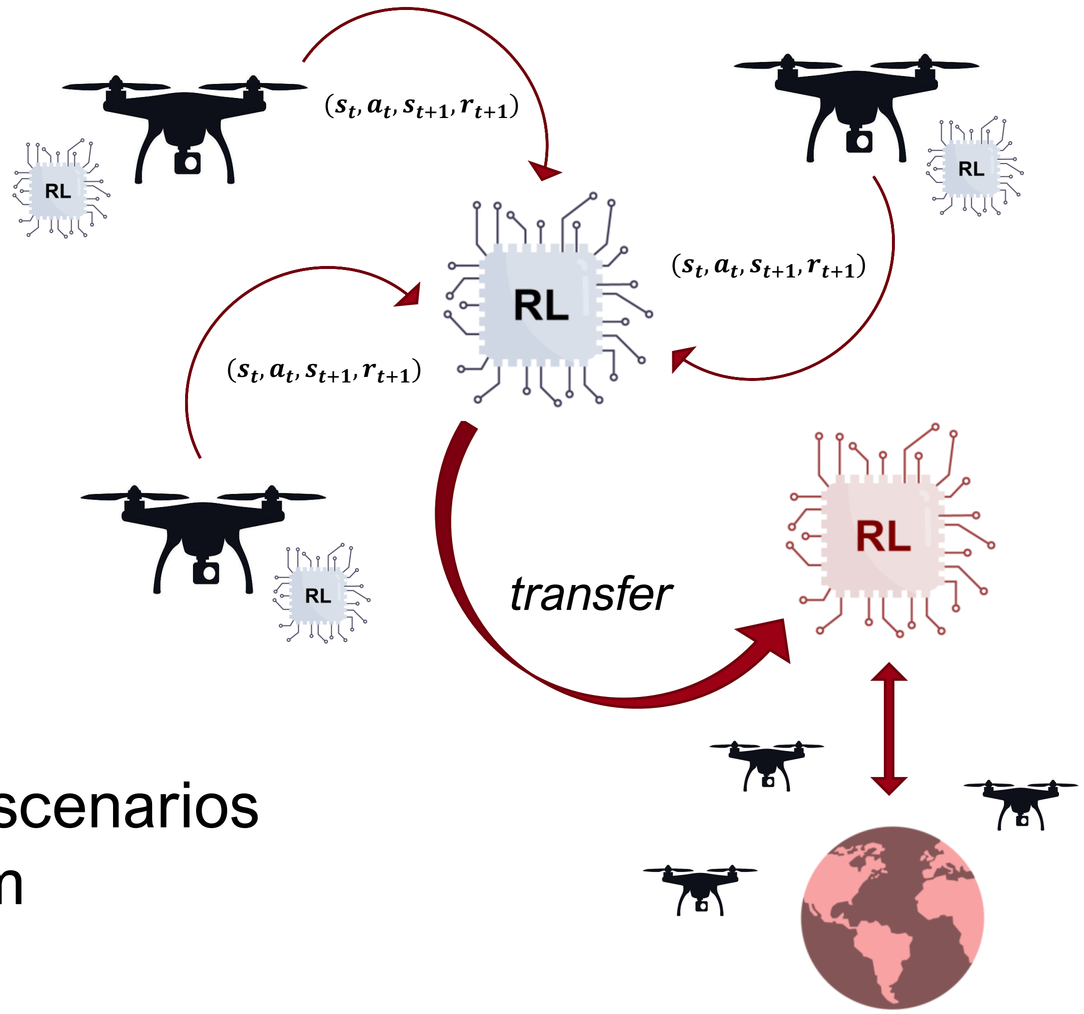
Success probability P_{succ} Step per episode N_{step}

Federico Mason - *masonfed@dei.unipd.it*

Distributed Learning: Approach

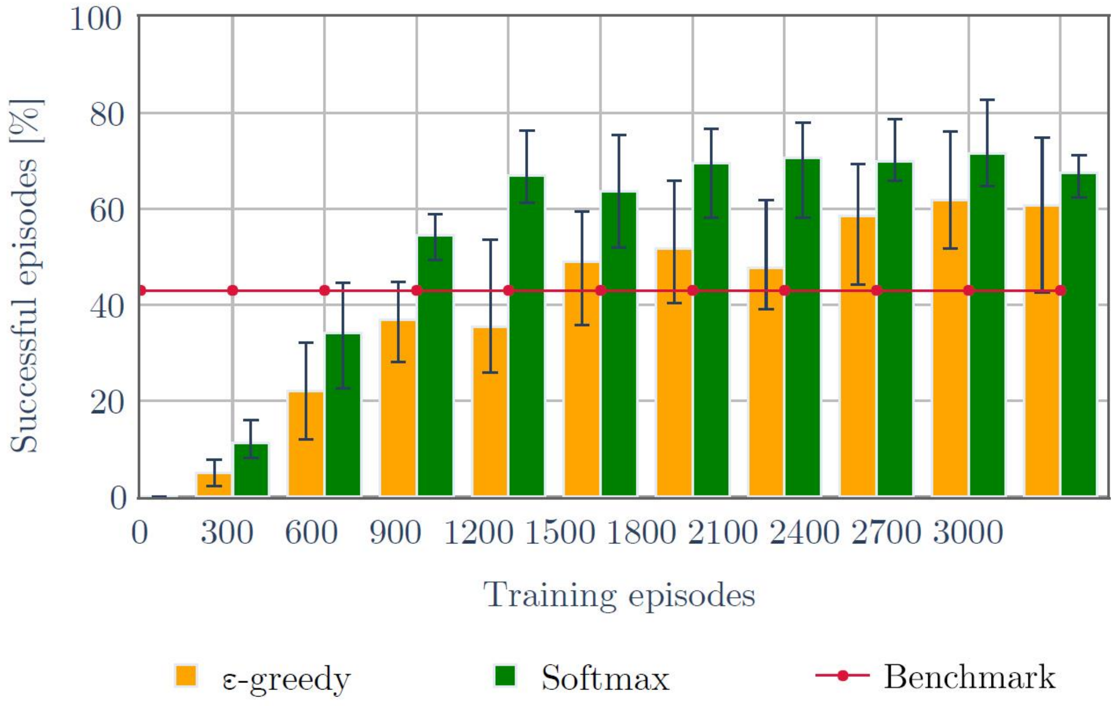
Centralized training:
all the nodes share their local experience with a central agent

Distributed implementation:
each node is controlled by a different instance of the central agent



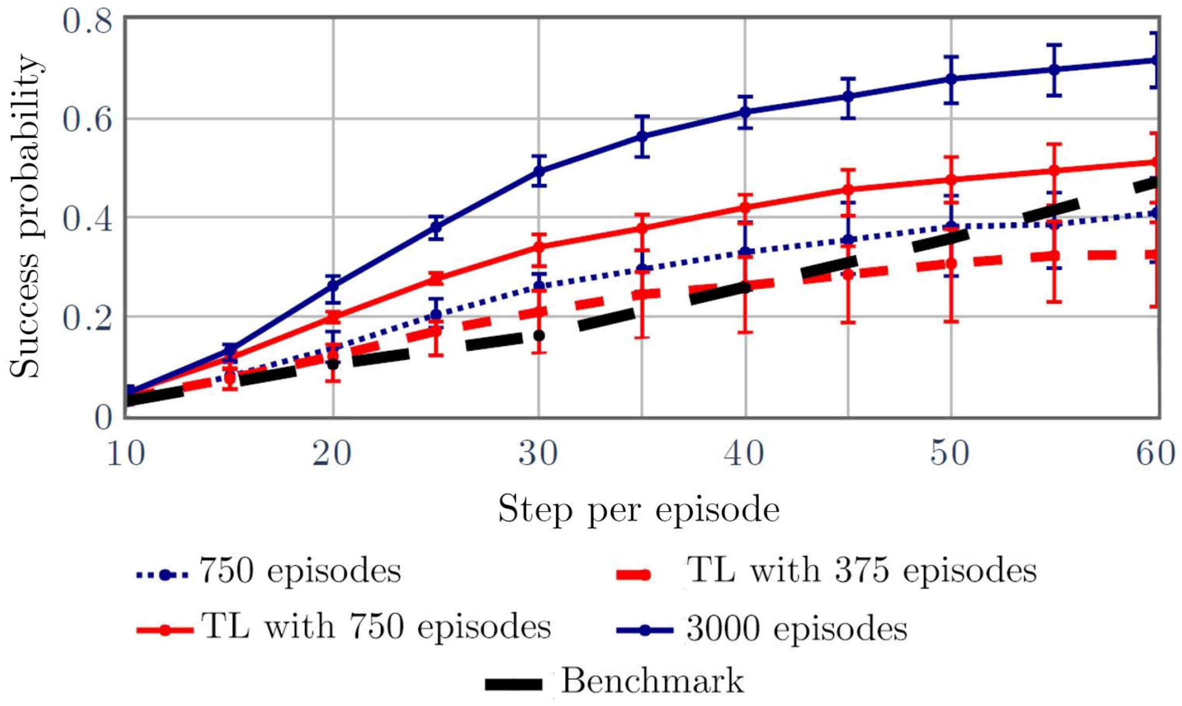
- We can adapt the same system to new scenarios by the **Transfer Learning (TL)** paradigm

Distributed Learning: Results



Benchmark → Look-ahead algorithm

- Ideal communication
- Realistic communication



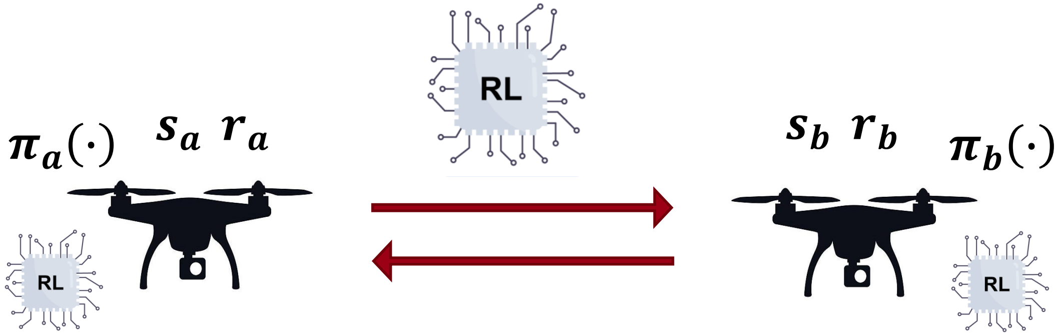
F. Venturini, F. Mason, F. Pase, F. Chiariotti, A. Zanella and M. Zorzi, "Distributed Reinforcement Learning for Flexible and Efficient UAV Swarm Control," in IEEE Transactions on Cognitive Communications and Networking, vol. 7, no. 3, pp. 955-969, September 2021

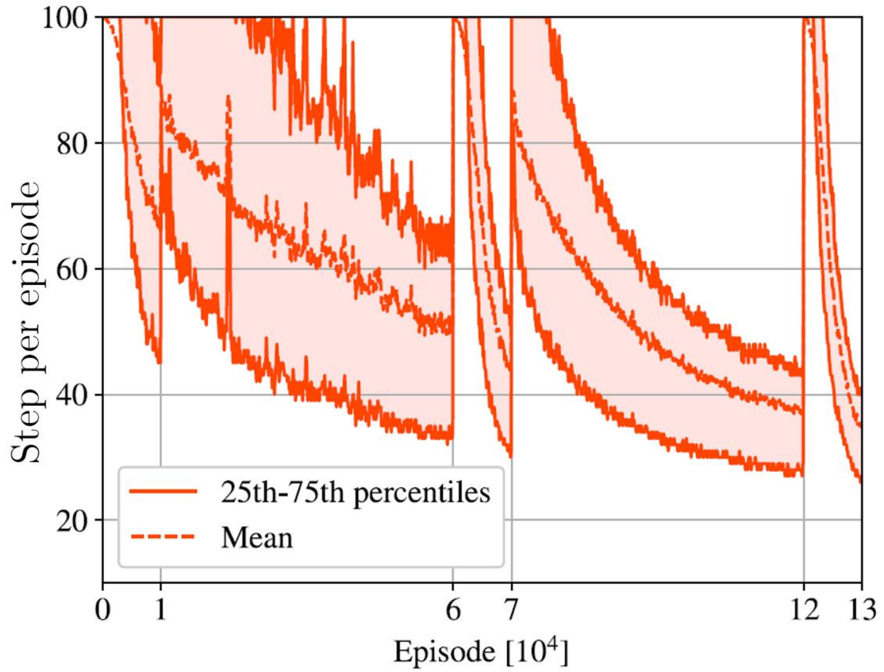
Distributed Learning: Approach

Ideal scenario:
all the nodes observes the same system state

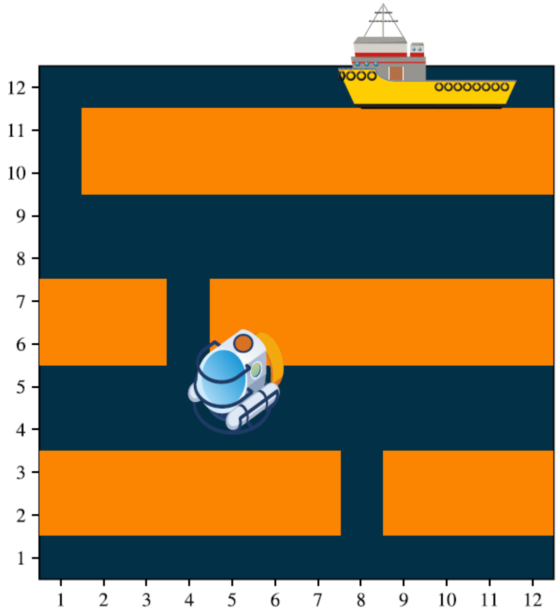
Benchmark scenario:
the communication policy is pre-determined

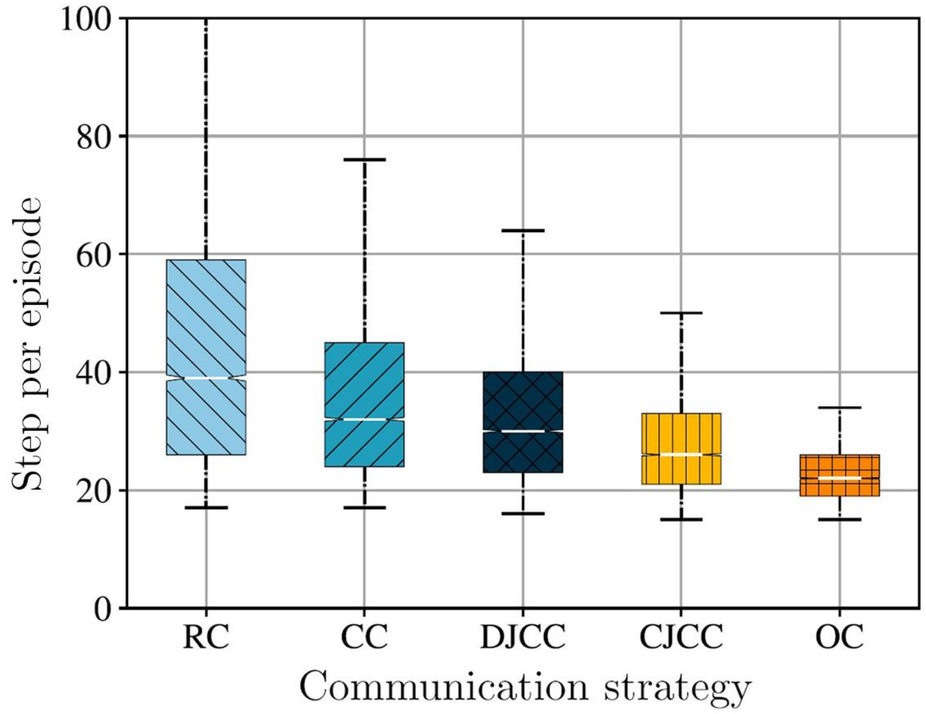
New proposal:
the communication policy is adapted to the control policy





Distributed Learning: Results





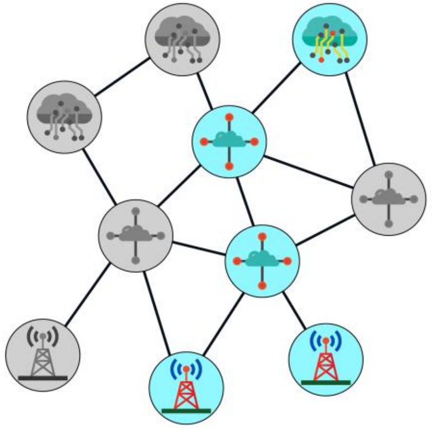
OC → Omniscient Communication
RC → Random Communication
CC → Closest Communication

DJCC → Distributed Joint Communication Control
CJCC → Centralized Joint Communication Control

Federico Mason - *masonfed@dei.unipd.it*

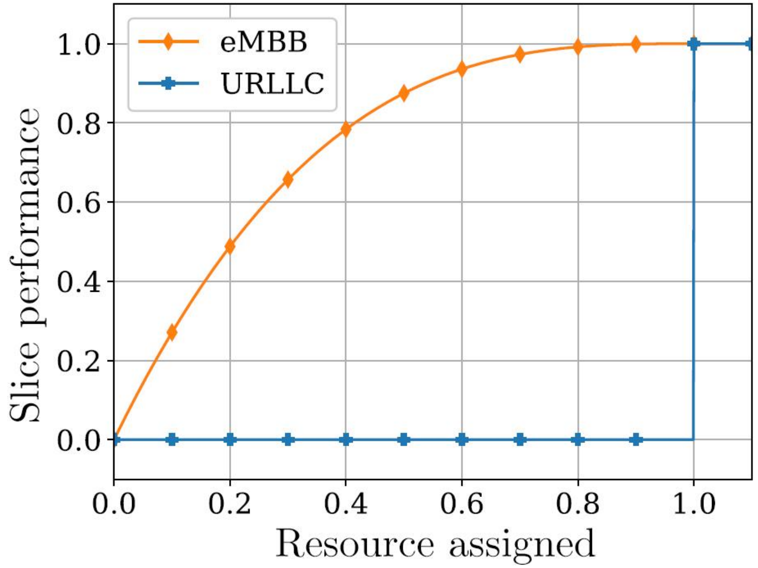
Resource Allocation: Scenario

We consider a network with multiple **traffic flows $\boldsymbol{\phi} \in \boldsymbol{\Phi}$** associated with specific **slice classes $\boldsymbol{\sigma} \in \boldsymbol{\Sigma}$**



- Flows contend for the same **computational** and **communication** resources

- Performance depends on both **resource allocation** and **slice characteristics**

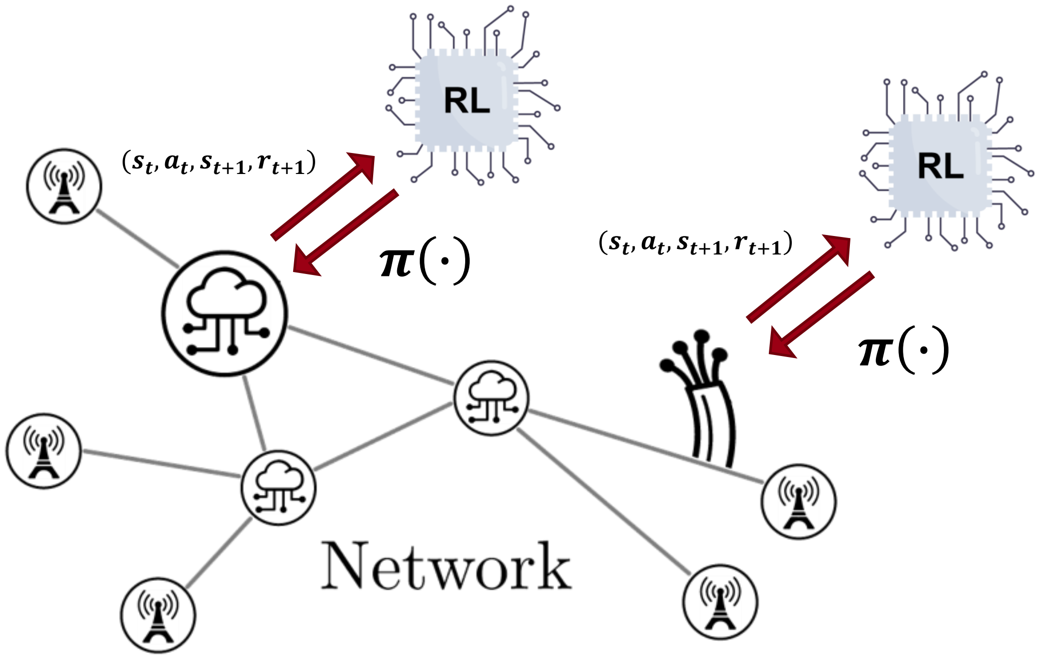


Federico Mason - *masonfed@dei.unipd.it*

Resource Allocation: Model

We associate each node $n \in \mathcal{N}$ and link $\ell \in \mathcal{L}$ with a **learning agent**. At each timeslot t:

- Each node $n \in \mathcal{N}$ allocate its **computational power**
- Each link $\ell \in \mathcal{L}$ allocates its **communication bandwidth**
- Each flow $\phi \in \Phi_\sigma$ computes its **performance** as $f_\sigma(\phi, t)$



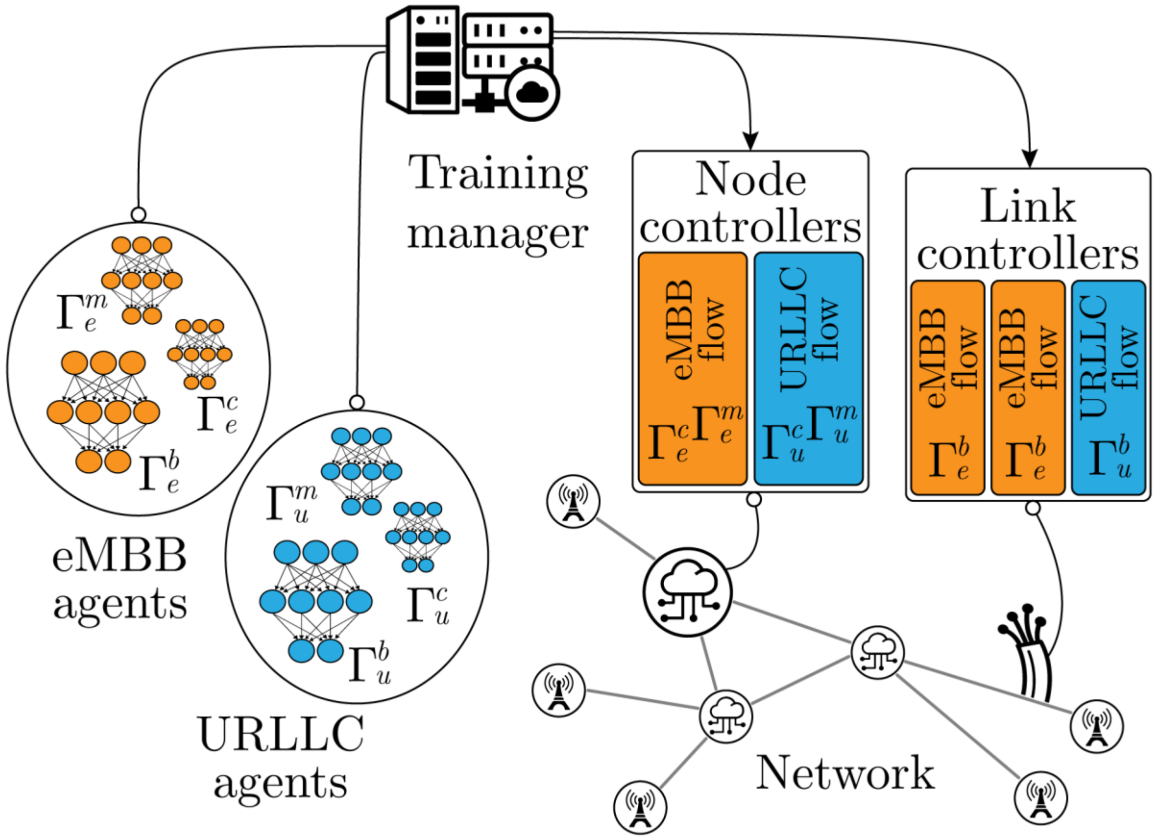
The **system performance** is given by:

$$\Omega = \frac{1}{|\Phi|} \sum_{\sigma \in \Sigma} \sum_{\phi \in \Phi_\sigma} f_\sigma(\phi, t)$$

Resource Allocation: Approach

Centralized training:
we train a different agent for each resource type and slice class

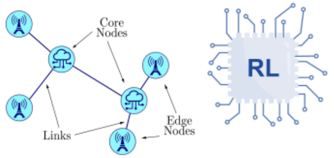
Distributed implementation:
we implement a different agent for each network facility and traffic flow



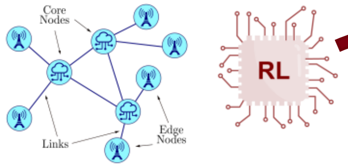
Transfer Learning (TL) can be used to refine the learned policy during the testing phase!

Resource Allocation: Results

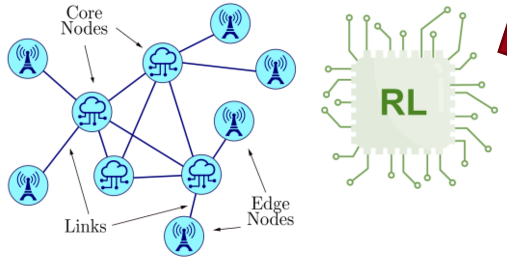
DRL-D

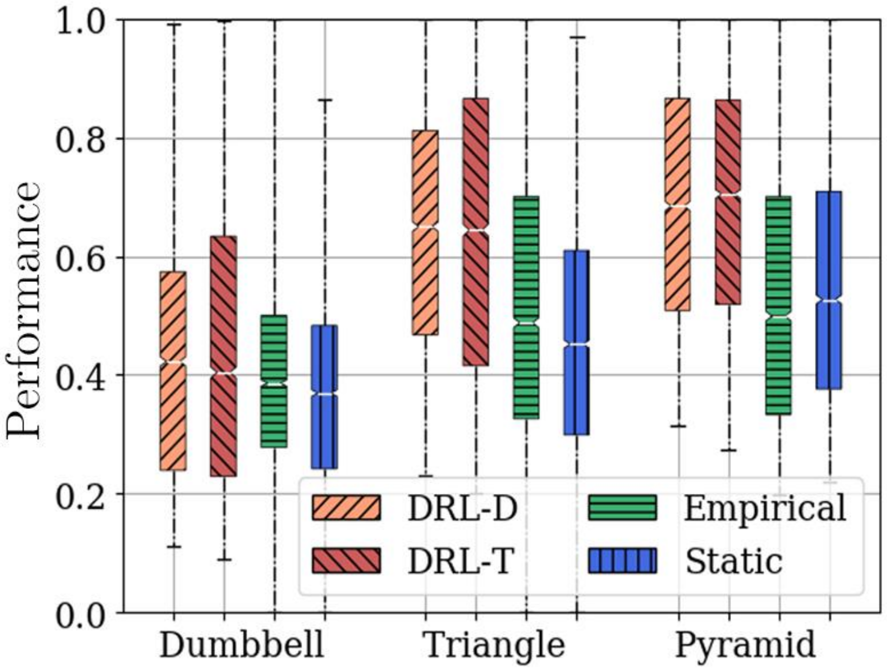


DRL-T

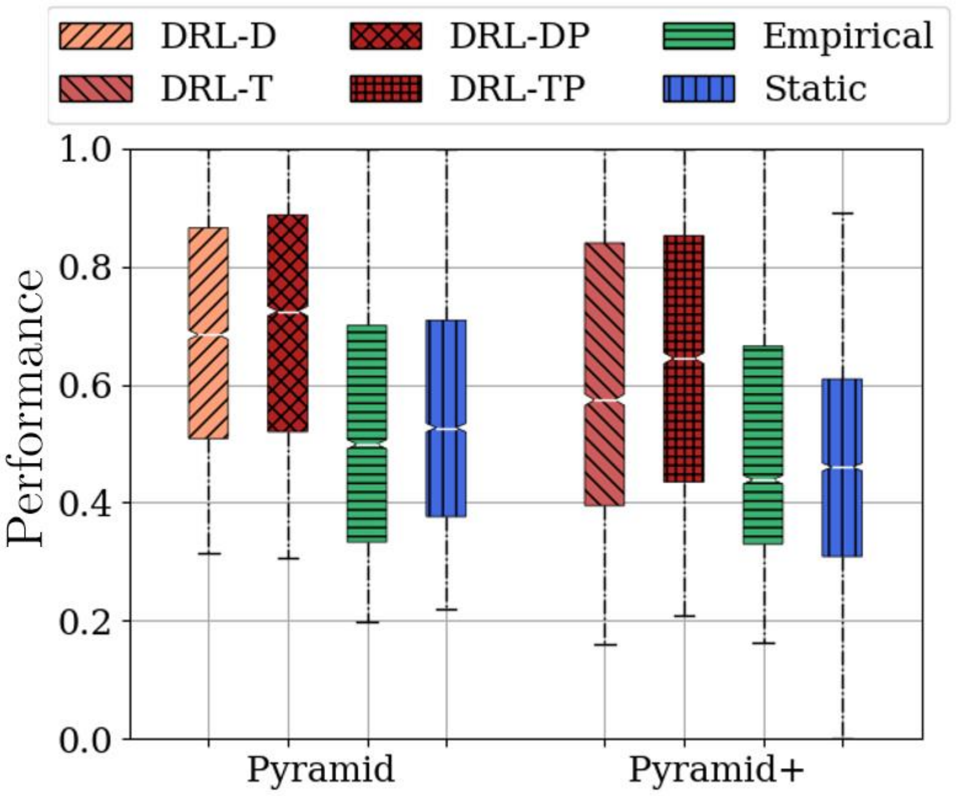


DRL-TP





DRL → Deep Reinforcement Learning

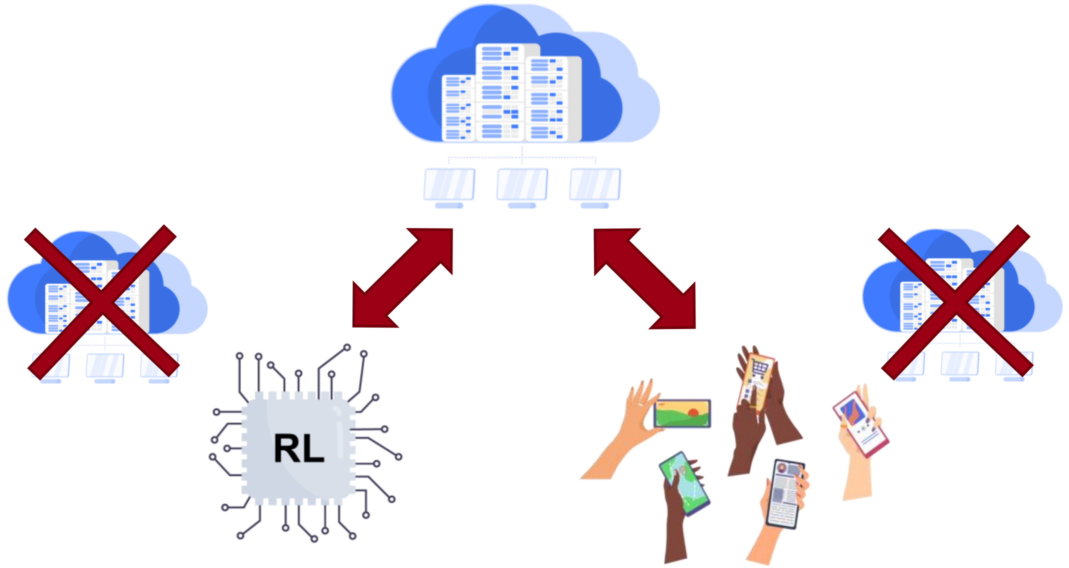


F. Mason, A. Zanella and G. Nencioni, “Using Distributed Reinforcement Learning for Resources Orchestration in a Network Slicing Scenario," in *IEEE Transactions on Networking*, vol. 31, no. 1, pp. 88-102, Feb. 2023.

Resource Allocation: Approach

Ideal scenario:
learning agents and network users exploit **dedicated** resources

Real scenario:
learning agents and network users exploit **shared** resources



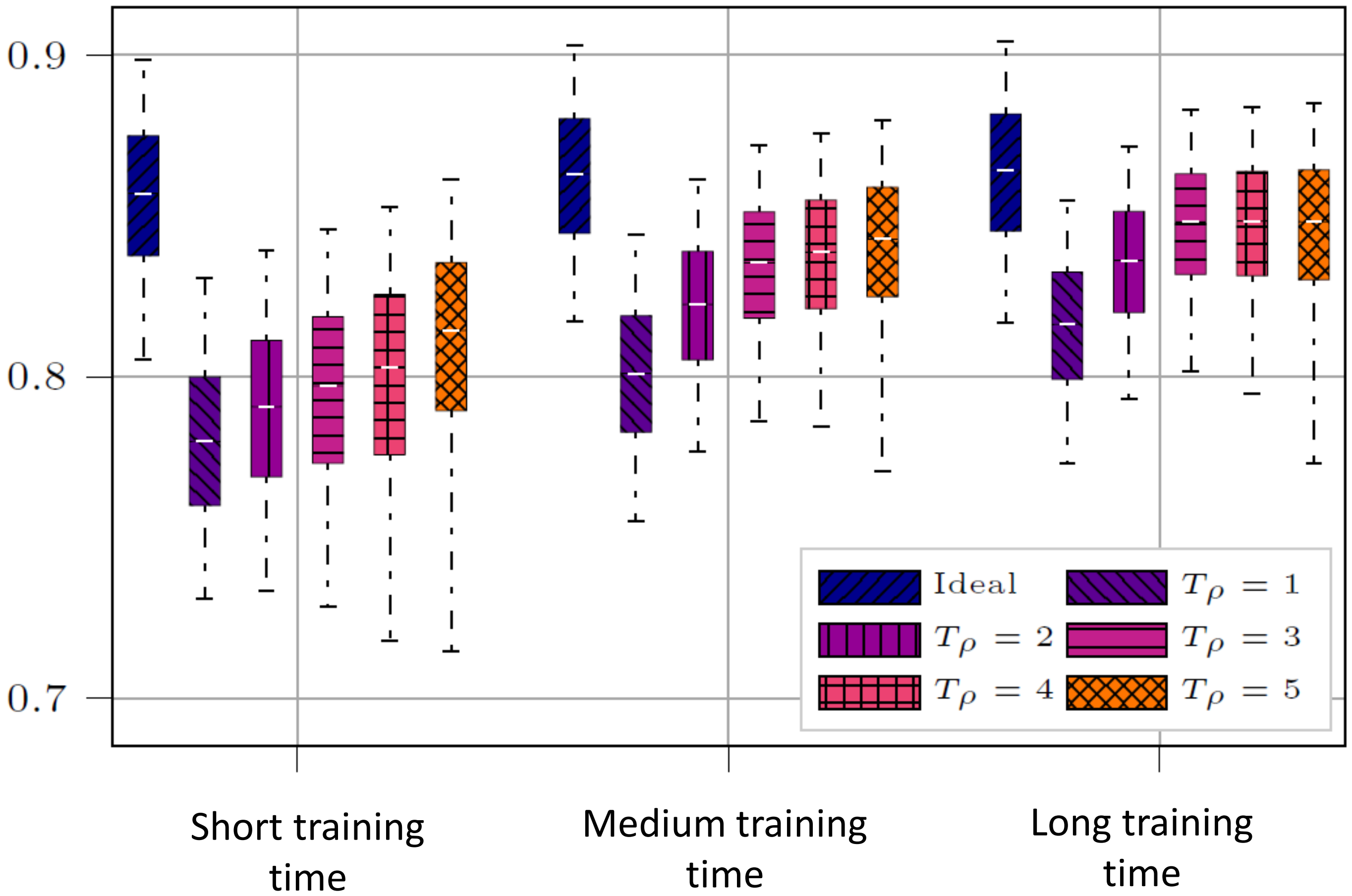
There is a **trade-off** between:

Immediate performance

Future adaptability

Resource Allocation: Results

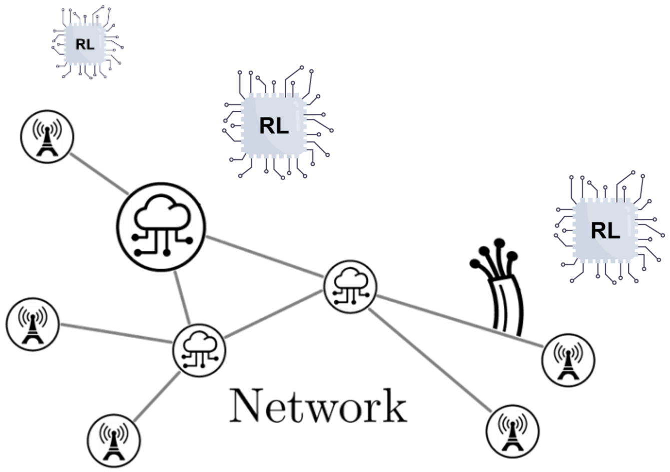
T_ρ → Percentage of training resources



F. Mason, F. Chiariotti, and A. Zanella, "No Free Lunch: Balancing Learning and Exploitation at the Network Edge," in *IEEE International Conference on Communications*, Seoul, South Korea, 2022

Conclusions

1. Networks require a more **flexible architecture** managed by multiple agents



2. **Communication, control**, and **resource allocation** must be jointly optimized

3. When **resources** are limited, **performance** is inversely proportional to **adaptability**

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

800 ANNI 1222·2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

SIGNET
signet.dei.unipd.it

Communication and Learning in Dynamic Networks

Federico Mason

Supervisor: Prof. Andrea Zanella

Email: masonfed@dei.unipd.it

GTTI annual meeting, September 11, 2023