Distributed Federated Learning and Optimization in Decision Systems and Cooperative Robotics

A. Camisa | A. Testa | I. Notarnicola | **G. Notarstefano**

Department of Electrical, Electronic, and Information Engineering
Alma Mater Studiorum Università di Bologna

giuseppe.notarstefano@unibo.it

OPT4SMART – ERC Project

Distributed Optimization Methods for Smart Cyber-Physical Networks

Methodological framework for distributed optimization

Numerical tools for machine learning and control

Experimental testbed and toolbox

opt4smart.dei.unibo.it

Computational Architectures

- Centralized methods relying on high-performance clusters

MARCONI Supercomputer
processing pwr: 18 PFLOPS

Computational Architectures

- Centralized methods relying on high-performance clusters

- Federated methods for large-scale problems
  with several computations coordinated by a master unit

Computational Architectures
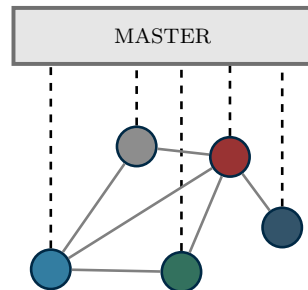
- Centralized methods relying on high-performance clusters

- Federated methods for large-scale problems
  with several computations coordinated by a master unit

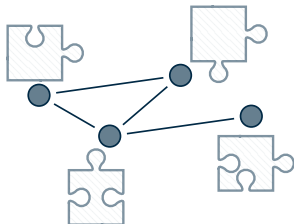- Distributed methods exploit only peer-to-peer communication
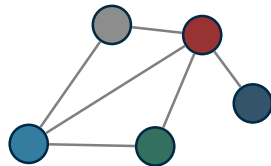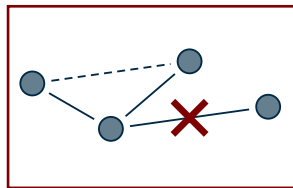
Distributed Optimization Paradigm
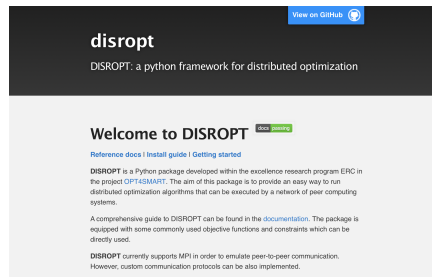
Optimization

$$\min_x \ f(x)$$
$$\text{subj.to} \ x \in X$$

Network

Problem data is spatially distributed and private

Exchange computation rather than data

DISROPT: Distributed Optimization Architecture

DISROPT

Toolbox for distributed optimization in python™

developed by OPT4SMART group
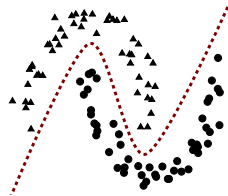

`https://opt4smart.github.io/disropt/`

Peer agents without a master

Heterogenous processors may work asynchronously

Communication may change over time
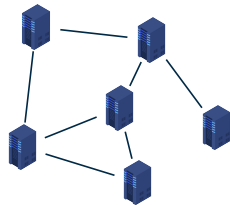
$$\min_x \; \sum_{i=1}^{N} f(x; u_i, y_i)$$

Paradigm

- local private data $u_i = (u_i^1, u_i^2, u_i^3)$ and y_i

- learn common parameters $x^\star \in \mathbb{R}^d$
  (common neural network)

- communication with neighbors only

- cooperate to learn from all data

Input layer   Hidden layer   Output layer

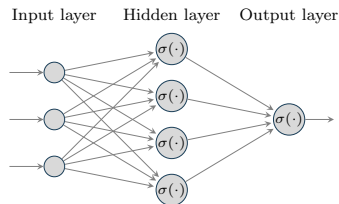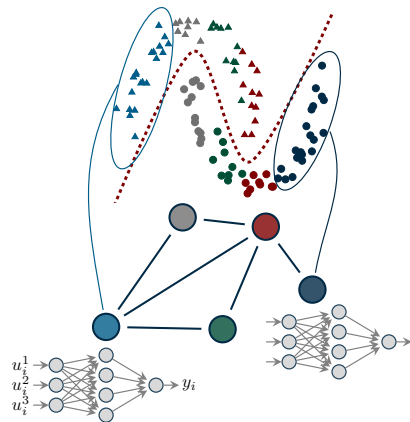# Distributed Federated Machine Learning

$$\min_x \ \sum_{i=1}^{N} f(x; u_i, y_i)$$

Paradigm

- local private data $u_i = (u_i^1, u_i^2, u_i^3)$ and y_i

- learn common parameters $x^\star \in \mathbb{R}^d$
  (common neural network)

- communication with neighbors only

- cooperate to learn from all data

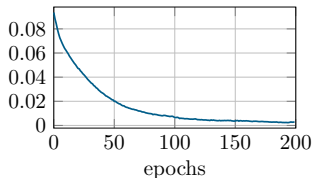# Large-scale Distributed Neural Network Training

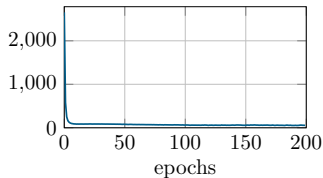100 agents communicating over a random graph
Neural network: ResNet50 with dataset CIFAR-10 (60k evenly shared color images)
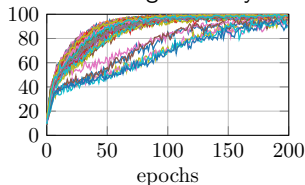Training on CINECA with GT-ADAM distributed algorithm + Disropt/Tensorflow
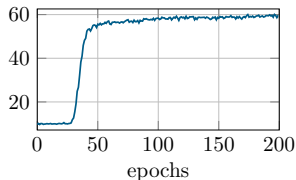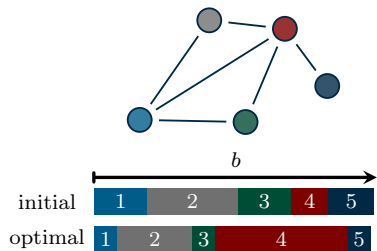

Training loss

Consensus error

Training accuracy

Test accuracy

$$\min_{x_1,\ldots,x_N} \sum_{i=1}^{N} f_i(x_i)$$

$$\text{subj. to } \sum_{i=1}^{N} g_i(x_i) \leq b$$

Paradigm

- negotiate a common resource b among agents

- optimize local decision x_i satisfying the limited budget

- communication with neighbors only

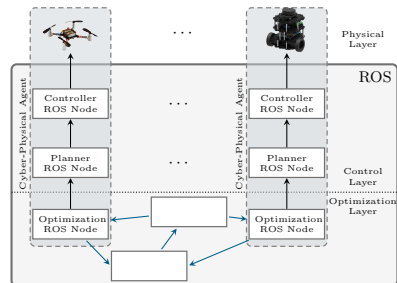# Distributed Cooperative Autonomous Robots

Team of N heterogeneous (mobile) robots executing complex tasks

Operation framework

Robot-to-robot communication

No central computation unit

Distributed Cooperative Autonomous Robots

Team of N heterogeneous (mobile) robots executing complex tasks

Operation framework
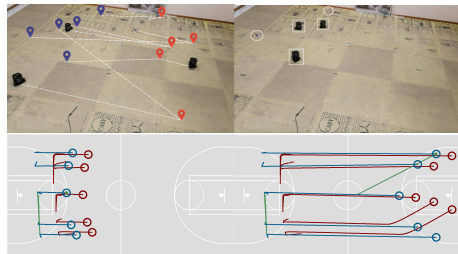
Robot-to-robot communication

No central computation unit

Complex tasks

task allocation, pickup & delivery

surveillance, patrolling, exploration
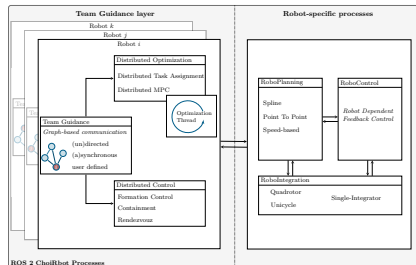
ChoiRbot: ROS 2 Toolbox for Cooperative Robotics

ChoiRbot

ROS 2 Toolbox for cooperative robotics

developed by OPT4SMART group ⠿2

Distributed robotics architecture

Distributed task allocation and decision-making

Distributed feedback laws (distributed MPC)
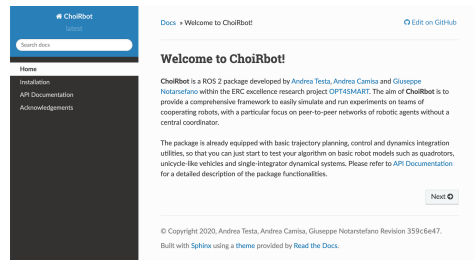
Enable distributed optimization over robots

ChoiRbot: ROS 2 Toolbox for Cooperative Robotics

ChoiRbot

ROS 2 Toolbox for cooperative robotics

developed by OPT4SMART group ⁝⁝ 2

https://github.com/OPT4SMART/ChoiRbot

Distributed task allocation and decision-making

Distributed feedback laws (distributed MPC)

Enable distributed optimization over robots

Sum Up

- Distributed optimization and learning

- Distributed cooperative robotics

- Toolbox and experiments for distributed computation