



Finanziato
dall'Unione europea
NextGenerationEU



Ministero
dell'Università
e della Ricerca



Italiadomani
PIANO NAZIONALE
DI RIPRESA E RESILIENZA



UNIVERSITÀ DEGLI STUDI
DI CAGLIARI



Robust Online Learning over Networks

N. Bastianello^(a) and D. Deplano^(b), M. Franceschelli^(b), K.H. Johansson^(c)

(a) Dept of Electronic and Electrical Engineering, University College London, United Kingdom.

(b) Department of Electrical and Electronic Engineering, University of Cagliari, Italy

(c) School of Electrical Engineering and Computer Science, and Digital Futures, KTH Royal Institute of Technology, Stockholm, Sweden

IEEE CSS Days – Session: “Meet the Awardees” – October 22, 2026

Outline

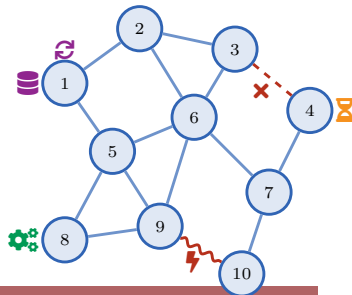
- ① Problem of interest and contributions
- ② Proposed algorithm (DOT-ADMM)
- ③ Convergence analysis and novelty w.r.t. the state-of-the-art
- ④ Numerical simulations
- ⑤ Extensions and applications

Problem of interest and contributions

Problem of interest

Agents cooperate to train a global model without sharing their local, private data, under challenging conditions:

- Local datasets change over time (online learning);
- Asynchronous processing/updates;
- Unreliable (faulty and noisy) communications;
- Inexact local computations.



Contributions

- DOT-ADMM: a distributed algorithm robust to all these challenges;
- Linear convergence to a neighborhood of the optimal solutions, whose size depends on the challenges;
- Numerical comparison: only DOT-ADMM is robust to all the challenges.

Outline

- ① Problem of interest and contributions
- ② Proposed algorithm (DOT-ADMM)
- ③ Convergence analysis and novelty w.r.t. the state-of-the-art
- ④ Numerical simulations
- ⑤ Extensions and applications

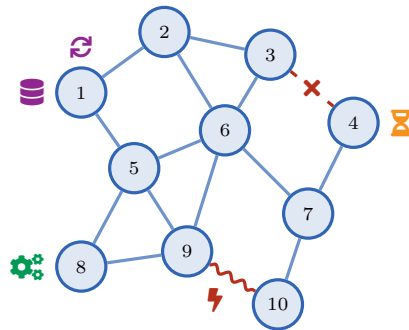
Formalization of the problem of interest

We consider n agents linked by an undirected, connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Each agent $i \in \mathcal{V}$ has a state $x_i(k) \in \mathbb{R}^p$ and a time-varying local cost $f_{i,k} : \mathbb{R}^p \rightarrow \mathbb{R} \cup \{+\infty\}$, $k \in \mathbb{N}$. The network aims at finding a common model that minimizes the sum of all local costs, as formulated next:

$$\begin{aligned} \min_{x_i} \quad & \sum_{i \in \mathcal{V}} f_{i,k}(x_i) \\ \text{s.t.} \quad & x_i = x_j \text{ if } (i, j) \in \mathcal{E} \end{aligned} \quad (1)$$

whose set of solutions is denoted by $\mathcal{X}_k^*$, in the following challenging scenario:

-  Local datasets change over time (online learning);
-  Asynchronous processing/updates;
-  Unreliable (faulty and noisy) communications;
-  Inexact local computations.



Formalization of the problem of interest: assumptions on the challenging scenario

Assumption 1 (local costs)

At each time $k \in \mathbb{N}$, the local cost functions $f_{i,k}$ are proper, lower semi-continuous, and convex.

Assumption 2 (local datasets change over time)

 The set of (primal-dual) solutions is non-empty at each time $k \in \mathbb{N}$. In addition, every solution at time $k - 1$ is at distance at most $\sigma \geq 0$ from the set of solutions at time k .

Assumption 3 (asynchronous updates and faulty communications)

 At each time step, each node $j \in \mathcal{V}$ has completed its local computation and successfully transmits data to its neighbors $i \in \mathcal{N}_j$ with probability $p_{ij} \in (0, 1]$, independently over time.

Assumption 4 (inexact local computations and noisy communications)

 Each node $i \in \mathcal{V}$ updates its local variable and receives information from neighbors $j \in \mathcal{N}_i$ with an additive noise such that the overall error $e_{ij}(k)$ is bounded in expectation by $\mathbb{E} [\|e_{ij}(k)\|] \leq \nu_e < \infty$.

Formalization of the problem of interest: assumptions on the challenging scenario

Assumption 1 (local costs)

At each time $k \in \mathbb{N}$, the local cost functions $f_{i,k}$ are proper, lower semi-continuous, and convex.

Assumption 2 (local datasets change over time)

 The set of (primal-dual) solutions is non-empty at each time $k \in \mathbb{N}$. In addition, every solution at time $k - 1$ is at distance at most $\sigma \geq 0$ from the set of solutions at time k .

Assumption 3 (asynchronous updates and faulty communications)

 At each time step, each node $j \in \mathcal{V}$ has completed its local computation and successfully transmits data to its neighbors $i \in \mathcal{N}_j$ with probability $p_{ij} \in (0, 1]$, independently over time.

Assumption 4 (inexact local computations and noisy communications)

 Each node $i \in \mathcal{V}$ updates its local variable and receives information from neighbors $j \in \mathcal{N}_i$ with an additive noise such that the overall error $e_{ij}(k)$ is bounded in expectation by $\mathbb{E} [\|e_{ij}(k)\|] \leq \nu_e < \infty$.

Formalization of the problem of interest: assumptions on the challenging scenario

Assumption 1 (local costs)

At each time $k \in \mathbb{N}$, the local cost functions $f_{i,k}$ are proper, lower semi-continuous, and convex.

Assumption 2 (local datasets change over time)

 The set of (primal-dual) solutions is non-empty at each time $k \in \mathbb{N}$. In addition, every solution at time $k - 1$ is at distance at most $\sigma \geq 0$ from the set of solutions at time k .

Assumption 3 (asynchronous updates and faulty communications)

 At each time step, each node $j \in \mathcal{V}$ has completed its local computation and successfully transmits data to its neighbors $i \in \mathcal{N}_j$ with probability $p_{ij} \in (0, 1]$, independently over time.

Assumption 4 (inexact local computations and noisy communications)

 Each node $i \in \mathcal{V}$ updates its local variable and receives information from neighbors $j \in \mathcal{N}_i$ with an additive noise such that the overall error $e_{ij}(k)$ is bounded in expectation by $\mathbb{E} [\|e_{ij}(k)\|] \leq \nu_e < \infty$.

Formalization of the problem of interest: assumptions on the challenging scenario

Assumption 1 (local costs)

At each time $k \in \mathbb{N}$, the local cost functions $f_{i,k}$ are proper, lower semi-continuous, and convex.

Assumption 2 (local datasets change over time)

 The set of (primal-dual) solutions is non-empty at each time $k \in \mathbb{N}$. In addition, every solution at time $k - 1$ is at distance at most $\sigma \geq 0$ from the set of solutions at time k .

Assumption 3 (asynchronous updates and faulty communications)

 At each time step, each node $j \in \mathcal{V}$ has completed its local computation and successfully transmits data to its neighbors $i \in \mathcal{N}_j$ with probability $p_{ij} \in (0, 1]$, independently over time.

Assumption 4 (inexact local computations and noisy communications)

 Each node $i \in \mathcal{V}$ updates its local variable and receives information from neighbors $j \in \mathcal{N}_i$ with an additive noise such that the overall error $e_{ij}(k)$ is bounded in expectation by $\mathbb{E} [\|e_{ij}(k)\|] \leq \nu_e < \infty$.

The Distributed Operator Theoretical (DOT) ADMM Algorithm

Input: For each agent $i \in \mathcal{V}$ initialize the auxiliary variables $\{z_{ij}(0)\}_{j \in \mathcal{N}_i}$; choose the relaxation $\alpha \in (0, 1)$ and the penalty $\rho > 0$.

Output: Each agent returns $x_i(k)$ that is an (approximated) solution to the distributed optimization problem in eq. (1).

for $k = 1, 2, \dots$ **each active agent** $i \in \mathcal{V}$

// asynchronous computations ⌚ and inexact updates ⚙️

receives a local cost $f_{i,k}$ 📦 and applies the local update

$$x_i(k) = \text{prox}_{f_{i,k}}^{1/\rho\eta_i} \left(\frac{1}{\rho\eta_i} \sum_{j \in \mathcal{N}_i} z_{ij}(k-1) \right) \quad (2)$$

for each agent $j \in \mathcal{N}_i$
transmits the packet

$$y_{i \rightarrow j}(k) = 2\rho x_i(k) - z_{ij}(k-1)$$

end for

for each packet $y_{j \rightarrow i}$ **received by agent** $j \in \mathcal{N}_i$

// noisy communications with packet loss ⚡

updates the auxiliary variable

$$z_{ij}(k) = (1 - \alpha)z_{ij}(k-1) + \alpha y_{j \rightarrow i}(k) \quad (3)$$

end for

end for

Outline

- 1 Problem of interest and contributions
- 2 Proposed algorithm (DOT-ADMM)
- 3 Convergence analysis and novelty w.r.t. the state-of-the-art**
- 4 Numerical simulations
- 5 Extensions and applications

Theorem 1 (Linear convergence)

Consider an online distributed optimization problem satisfying Assumptions 1-2, and a connected network of agents that solves it by running DOT-ADMM under Assumptions 3-4.

If the DOT-ADMM operator T_k , ruling the updates of the auxiliary variable

$$z(k) = T_k(z(k-1))$$

is γ -metric subregular, then there is an upper bound to the distance between the current solution $x(k)$ and the set of optimal solutions $\mathcal{X}_k^*$ that holds in mean for all $k \in \mathbb{N}$, and this bound has a linearly decaying dependence on the initial condition:

$$\mathbb{E} \left[d_{\mathcal{X}_k^*}(x(k)) \right] = O \left(\mu^k d_{T_0}(z(0)) + \frac{1 - \mu^k}{1 - \mu} (\nu_e + \sigma) \right), \quad \mu = \sqrt{1 - \frac{(1 - \alpha)\underline{p}}{\alpha\lambda}} \in (0, 1), \quad (4)$$

where $\underline{p} = \min_{(i,j) \in \mathcal{E}} p_{ij}$ and $\lambda > \max\{\gamma^2, (1 - \alpha)\underline{p}/\alpha\}$, so that there is an asymptotic upper bound that does not depend on the initial condition $z(0)$:

$$\limsup_{k \rightarrow +\infty} \mathbb{E} \left[d_{\mathcal{X}_k^*}(x(k)) \right] = O \left(\frac{\nu_e + \sigma}{1 - \mu} \right). \quad (5)$$

Metric subregularity

Definition 1

An operator $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is metric subregular if there is a positive constant $\gamma > 0$ such that

$$d_T(x) \leq \gamma \|(\text{Id} - T)x\|, \quad \forall x \in \mathbb{R}^n. \quad (6)$$

When γ is known, T is said to be γ -metric subregular.

Remarks:

- Metric subregularity of the operator is sufficient for linear convergence;
- Strong convexity of the local cost is neither necessary nor sufficient for linear convergence:
 - **Non necessity:** due to Theorem 1, a simple example is linear regression problems with rank-deficient data;
 - **Non sufficiency:** The strongly convex costs $f_i(x_i) = \sqrt[3]{x_i^4} + \frac{\varepsilon}{2}x_i^2$ yield a sub-linear convergence
- Metric subregularity holds for strongly convex local costs with Lipschitz continuous gradients.

Metric subregularity

Definition 1

An operator $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is metric subregular if there is a positive constant $\gamma > 0$ such that

$$d_T(x) \leq \gamma \|(\text{Id} - T)x\|, \quad \forall x \in \mathbb{R}^n. \quad (6)$$

When γ is known, T is said to be γ -metric subregular.

Remarks:

- Metric subregularity of the operator is sufficient for linear convergence;
- Strong convexity of the local cost is neither necessary nor sufficient for linear convergence:
 - **Non necessity:** due to Theorem 1, a simple example is linear regression problems with rank-deficient data;
 - **Non sufficiency:** The strongly convex costs $f_i(x_i) = \sqrt[3]{x_i^4} + \frac{\varepsilon}{2}x_i^2$ yield a sub-linear convergence
- Metric subregularity holds for strongly convex local costs with Lipschitz continuous gradients.

Corollary 1 (Particular cases)

Consider the scenario of Theorem 1 and the following simplified cases:

- (a) the cost functions $f_{i,k} = f_i$ are static, i.e., $\sigma = 0$;
- (b) communications are noiseless and computations are exact, i.e., there are no additive errors $\nu_e = 0$;
- (c) communications are synchronous.

Then the results of Theorem 1 become:

- i) (a) implies that the distance $d_{\mathcal{X}_k^*}(x(k))$ converges linearly to $O(\nu_e)$ in mean;
- ii) (b) implies that the distance $d_{\mathcal{X}_k^*}(x(k))$ converges linearly to $O(\sigma)$ in mean;
- iii) (a) $\wedge$ (b) implies that the distance $d_{\mathcal{X}^*}(x(k))$ converges linearly to zero in mean square with rate μ^2 ; moreover, $x(k)$ almost surely converges to the set of solutions $\mathcal{X}^*$;
- iv) (a) $\wedge$ (b) $\wedge$ (c) implies that the iteration is deterministic and the distance $d_{\mathcal{X}^*}(x(k))$ converges linearly to zero with rate μ .

Outline

- ① Problem of interest and contributions
- ② Proposed algorithm (DOT-ADMM)
- ③ Convergence analysis and novelty w.r.t. the state-of-the-art
- ④ Numerical simulations**
- ⑤ Extensions and applications

Set-up

The considered set-up is a network of $N = 10$ nodes, exchanging information through a random graph of 20 edges, that want to solve an online logistic regression problem with regularized local costs

$$f_{i,k}(x) = \sum_{h=1}^{m_i} \log(1 + \exp(-b_{i,h,k} a_{i,h,k} x)) + \frac{\varepsilon}{2} \|x\|^2$$

where $x \in \mathbb{R}^p$, $p = 16$, is the vector of weights and intercept to be learned, and $\{(a_{i,h,k}, b_{i,h,k})_{h=1}^{m_i}\}$ with $a_{i,h,k} \in \mathbb{R}^{1 \times p}$, $b_{i,h,k} \in \{-1, 1\}$ are the $m_i = 20$ feature vectors and class pairs at time $k \in \mathbb{N}$, and $\varepsilon = 5$.

Inexact local updates

N.B.: the proximal of $f_{i,k}$ does not have a closed form! Thus, each node computes an inexact solution via accelerated gradient descent, terminating when the distance between consecutive iterates is smaller than a threshold $\theta > 0$.

Table: Computational time of local updates and asymptotic error for a static logistic regression problem.

Threshold	Comp. time [s]	Asymptotic err.
$\theta = 10^{-14}$	3.47×10^{-3}	4.14×10^{-14}
$\theta = 10^{-12}$	2.84×10^{-3}	3.65×10^{-12}
$\theta = 10^{-10}$	2.42×10^{-3}	4.88×10^{-10}
$\theta = 10^{-8}$	1.95×10^{-3}	5.30×10^{-8}
$\theta = 10^{-6}$	1.39×10^{-3}	1.01×10^{-5}
$\theta = 10^{-4}$	8.88×10^{-4}	5.73×10^{-4}
$\theta = 10^{-2}$	4.12×10^{-4}	9.71×10^{-2}

Remarks:

- The inexact computation of the proximal introduces an additive error on the local update $x_i(k)$ (cf. Assumption 4);
- Smaller thresholds θ yield smaller asymptotic errors, which scale with θ , in line with the bound of Theorem 1;
- Smaller thresholds require longer computational times: θ trades accuracy for computation.

Quantized communications ⚡

Assume that each agent completes the local update using $\theta = 10^{-8}$ and then sends the quantized version $q(x)$ of a message $x \in \mathbb{R}^p$, namely

$$[q(x)]_j = \begin{cases} \underline{q} & \text{if } [x]_j < \underline{q} \\ \delta \lfloor [x]_j / \delta \rfloor & \text{if } \underline{q} \leq [x]_j \leq \bar{q}, \\ \bar{q} & \text{if } [x]_j > \bar{q} \end{cases}, \quad j \in \{1, \dots, p\}$$

with $\bar{q} = -\underline{q} = 10$, and $\delta > 0$ being the quantization level.

Table: Asymptotic error for different quantization levels.

Quantization	Asymptotic error
No quantization	5.30×10^{-8}
$\delta = 10^{-10}$	5.30×10^{-8}
$\delta = 10^{-8}$	7.36×10^{-8}
$\delta = 10^{-6}$	4.74×10^{-6}
$\delta = 10^{-4}$	5.64×10^{-4}
$\delta = 10^{-2}$	5.32×10^{-2}
$\delta = 10^{-1}$	4.91×10^{-1}

Remarks:

- Quantization acts as an additive noise on the transmitted packets (cf. Assumption 4);
- The asymptotic error scales with the quantization level δ , in line with the bound of Theorem 1;
- For $\delta \leq 10^{-8}$, the error is dominated by the inexact local updates with $\theta = 10^{-8}$.

Asynchronous updates ⌚

Assume that each agent completes the local update using $\theta = 10^{-8}$ with some probability $p \in \{\underline{p}, \bar{p}\}$ with $\underline{p} < \bar{p}$. The agents characterized by the smaller probability $\underline{p}$ are the “slow” nodes, which, having fewer computational resources, take on average a longer time to reach the threshold θ . Let N_s be the number of slow nodes.

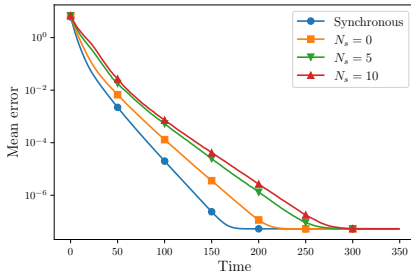


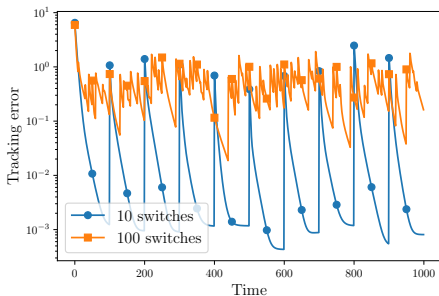
Figure: Error of DOT-ADMM for different numbers N_s of slow nodes.

Remarks:

- Slow agents complete their local updates with smaller probability, which results in random updates (cf. Assumption 3);
- More slow nodes imply less frequent updates and a slower linear convergence, in line with the convergence rate of Theorem 1;
- The asymptotic error is not affected by asynchrony, being determined by the inexact local updates.

Online optimization

The local costs are piece-wise constant: they change 10 and 100 times, respectively, and are generated so that the maximum distance between consecutive optima is ~ 2.5 . The local updates are computed with threshold $\theta = 10^{-4}$.



Remarks:

- In the transients, the convergence is linear, as predicted by Theorem 1 (cf. Assumption 2);
- When the problem changes less frequently, DOT-ADMM has time to converge to smaller errors, up to the bound imposed by the inexact local updates;
- More frequent changes yield larger tracking errors overall.

Figure: Tracking error of DOT-ADMM with piece-wise constant costs.

Comparison with state-of-the-art algorithms

We compare DOT-ADMM with three gradient-based methods, for both static and online problems:

- *ra-GD*: gradient tracking algorithm which makes use of the robust ratio consensus to ensure convergence in the presence of asynchrony.

Bof, N., Carli, R., Notarstefano, G., Schenato, L., and Varagnolo, D. (2018). Multiagent Newton–Raphson optimization over lossy networks. *IEEE Transactions on Automatic Control*, 64(7), 2983-2990.

- *LEAD*: gradient tracking algorithm which is designed to be robust to a certain class of unbiased quantizers.

Liu, X., Li, Y., Wang, R., Tang, J., and Yan, M. (2020). Linear convergent decentralized optimization with compression. *arXiv preprint arXiv:2007.00232*.

- *DGD*: which does not converge exactly, but has been shown to be robust to additive errors and online scenarios.

Yuan, K., Ling, Q., and Yin, W. (2016). On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3), 1835-1854.

Fair comparison: since the local updates of DOT-ADMM require a longer time, *ra-GD*, *LEAD*, and *DGD* are run for a larger number of iterations to match its computational time; all step-sizes are hand-tuned for optimal performance.

Comparison with state-of-the-art algorithms: static problems

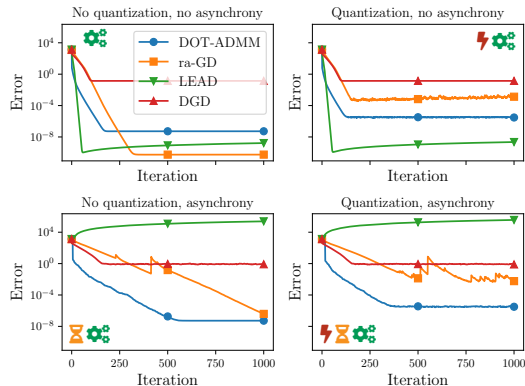
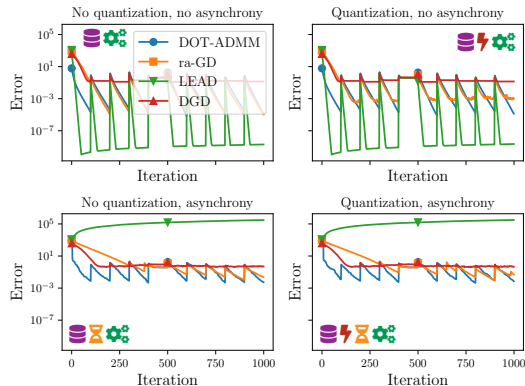


Figure: Comparison on static problems combining quantization/asynchrony.

Remarks:

- ra-GD is robust to asynchrony, but converges slowly due to a necessarily conservative step-size, and with quantization it reaches a larger neighborhood than DOT-ADMM (cf. Assumptions 3-4);
- LEAD converges with quantization, but it is not robust to asynchrony and seems to diverge when the agents are not synchronized;
- DGD is robust to both quantization and asynchrony, but its inherent inexactness leads to poor performance.

Comparison with state-of-the-art algorithms: online problems



Remarks:

- The local costs are piece-wise constant, changing every 100 iterations;
- While the costs do not change, the behavior of the algorithms is similar to that on static problems, with an increase of the error any time the problem changes (cf. Assumption 2);
- Only DOT-ADMM shows robustness to asynchrony, quantization, and time-varying costs.

Figure: Comparison on online problems combining quantization/asynchrony.

Outline

- ① Problem of interest and contributions
- ② Proposed algorithm (DOT-ADMM)
- ③ Convergence analysis and novelty w.r.t. the state-of-the-art
- ④ Numerical simulations
- ⑤ Extensions and applications**

This work:

- N. Bastianello and D. Deplano, M. Franceschelli, K.H. Johansson, “Robust online learning over networks”, IEEE Transactions on Automatic Control, 2025.

Application to dynamic consensus and personalized learning:

- D. Deplano, N. Bastianello, M. Franceschelli, K.H. Johansson, “A unified approach to solve the dynamic consensus on the average, maximum, and median values with linear convergence”, IEEE Conference on Decision and Control, 2023.
- D. Deplano, N. Bastianello, M. Franceschelli, K.H. Johansson, “Distributed variance consensus with application to personalized learning”, IEEE Conference on Decision and Control, 2023.

Extension to composite optimization problems:

- D. Deplano, N. Bastianello, M. Franceschelli, K.H. Johansson, “Composite optimization in open multi-agent systems under additive errors”, IEEE Control Systems Letters, 2026.

Extension to Open MAS:

- D. Deplano, N. Bastianello, M. Franceschelli, K.H. Johansson, “Optimization and learning in open multi-agent systems”, IEEE Transactions on Automatic Control, 2026.

Take-home messages

- DOT-ADMM is robust to time-varying data 🗄️, asynchronous computations ⌚, unreliable communications ⚡, and inexact local computations ⚙️;
- Metric subregularity of the DOT-ADMM operator, rather than strong convexity of the costs, is what yields linear convergence;
- The tracking error converges linearly in mean to a neighborhood of size $O\left(\frac{\nu_e + \sigma}{1 - \mu}\right)$, which vanishes for static costs without errors.



Finanziato
dall'Unione europea
NextGenerationEU



Ministero
dell'Università
e della Ricerca



Italiadomani
PIANO NAZIONALE
DI RIPARAZIONE E RESILIENZA



UNIVERSITÀ DEGLI STUDI
DI CAGLIARI



Robust Online Learning over Networks

Thank you for your attention!

Diego Deplano

Email: diego.deplano@unica.it

Webpage: <https://diegodeplano.github.io/>



SCAN ME