DYNAMIC RESOURCE OPTIMIZATION FOR ADAPTIVE FEDERATED LEARNING EMPOWERED BY RECONFIGURABLE INTELLIGENT SURFACES

Paper ID: 4643

C. BATTILORO[1] M. MERLUZZI[2] P. DI LORENZO[1] S. BARBAROSSA[1]

[1] SAPIENZA UNIVERSITY OF ROME, DEPARTMENT OF INFORMATION ENGINEERING, ELECTRONICS, AND TELECOMMUNICATIONS, VIA EUDOSSIANA 18, 00184, ROME, ITALY

[2] CEA-LETI, UNIVERSITE GRENOBLE ALPES, F-38000 GRENOBLE, FRANCE

ABSTRACT

The aim of this work is to propose a novel dynamic resource allocation strategy for data-driven, adaptive, energy-efficient Federated Learning (FL) with latency and learning performance guarantees, endowed with Reconfigurable Intelligent Surfaces (RISs).

Contribution: Novel dynamic optimization framework for adaptive federated learning endowed with RISs, jointly encompassing radio and computation aspects in order to strike the best trade-off between energy, latency, and performance of the federated learning task.

FEDERATED LEARNING TASK

- N edge devices and an AP equipped with an edge server
- Consider the learning problem in the unknown model variable $\boldsymbol{w}$

$$\min_{\mathbf{w}} \sum_{i=1}^{N} \mathbb{E}\{J_i(\mathbf{w}; \boldsymbol{x}_i, y_i)\}$$

- At each t, the edge devices compute $\nabla J_i(\mathbf{w}; \boldsymbol{x}_{i,t}, y_{i,t})$ over a batch of data $\mathcal{B}_t$ of size $|\mathcal{B}_t| = B_t$ and upload them to the AP
- The edge server computes $\mathbf{w}_{t+1}$ via *any* gradient-based algorithm and fed it back to the devices. In general:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \mu \cdot f\left(\sum_{i\in\mathcal{S}_t} \nabla J_i(\mathbf{w}; \boldsymbol{x}_{i,t}, y_{i,t})\right)$$

RIS-ENHANCED COMMUNICATIONS

- K passive RISs with M reflecting elements
- The phase of each element is quantized using b_r bits
- Each element has a complex reflection coefficient:

$$v_{k,l,t} \in \mathcal{R} = \left[0, \left\{e^{j\frac{2n\pi}{2^{b_r}}}\right\}_{n=0}^{2^{b_r}-1}\right], \quad \forall k, l, t$$

- The RIS-aided uplink transmission rate between user i and the AP:

$$R_{i,t} = B_i \log_2\left(1 + \frac{h_{i,t}(\boldsymbol{v}_t)p_{i,t}}{N_0 B_i}\right),$$

where $h_{i,t}(\boldsymbol{v}_t)$ is the RIS-dependent channel coefficient:

$$h_{i,t}(\boldsymbol{v}_t) = \left|h^a_{i,t} + \sum_{k=1}^{K} \boldsymbol{h}^T_{i,k,t}\,\mathrm{diag}(\boldsymbol{v}_{k,t})\,\boldsymbol{z}^a_{i,k,t}\right|^2$$

LATENCY OF TRAINING ITERATIONS

- Local processing time: $L^{loc}_{i,t} = \dfrac{B_t J_i}{f_{i,t}}$, where f^l_i is the local CPU frequency
- Uplink communication time: $L^u_{i,t} = \dfrac{m \cdot b_{i,t}}{R_{i,t}}$, where $R_{i,t}$ is the uplink data rate.
- Remote processing time: $L^s_t = \dfrac{C|\mathcal{S}_t|}{f^s_t}$, where f^s is the remote frequency of the server.

The overall latency at time t is given by:

$$L_t = \max_{i\in\mathcal{S}_t}\left\{L^{loc}_{i,t} + L^u_{i,t}\right\} + L^s_t$$

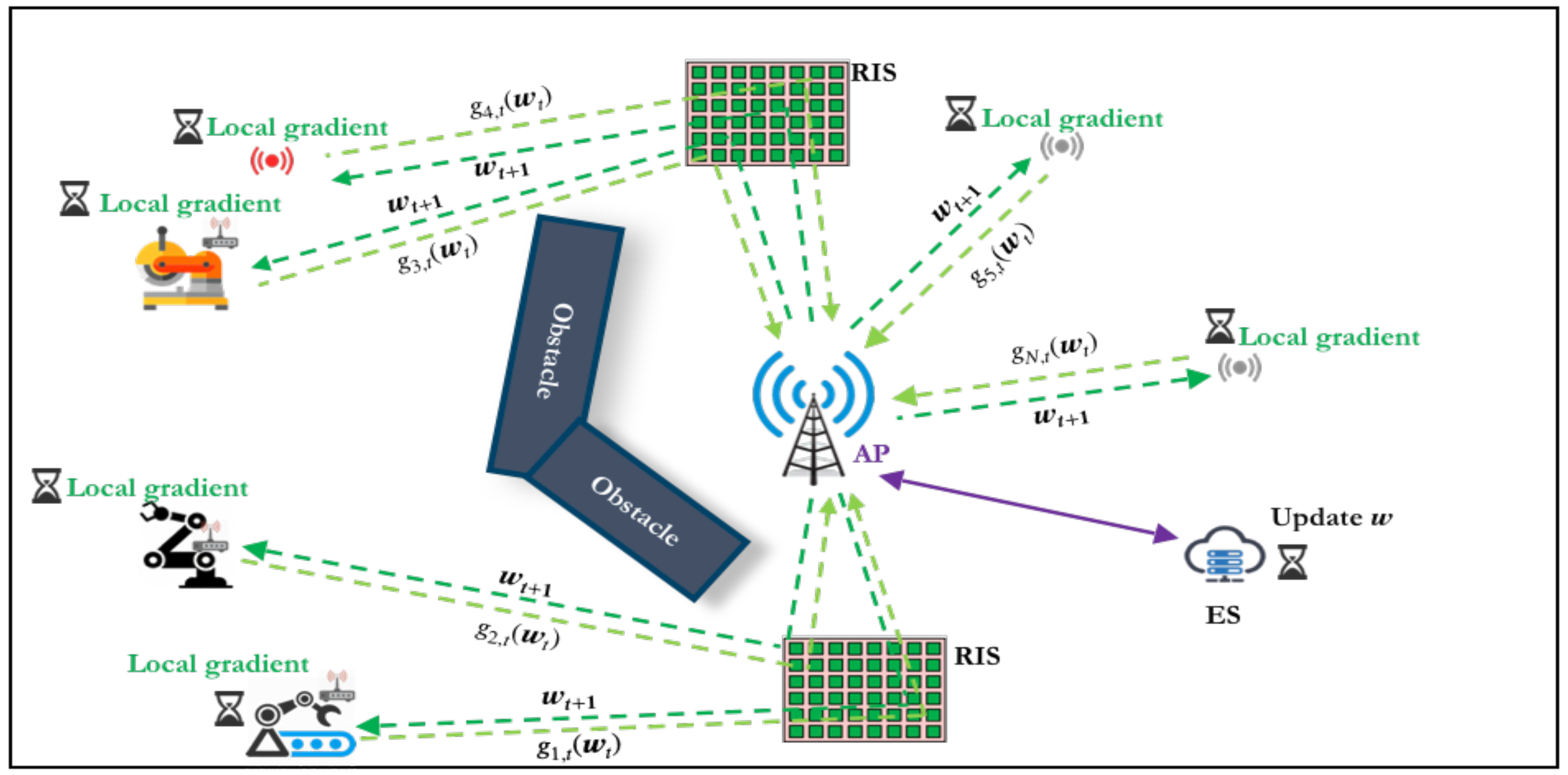
POWER CONSUMPTION

- Power spent for local computation [1]: $p^c_{i,t} = \gamma_l(f_{i,t})^3$
- Power spent for uplink transmission: $p_{i,t} = \dfrac{B_i N_0}{h_{i,t}}\left[\exp\left(\dfrac{R_{i,t}\ln 2}{B^u_i}\right) - 1\right]$
- Power spent for remote computation: $p^c_{s,t} = \gamma_r(f^s_t)^3$

The overall power consumption at time t is given by:

$$p^{\text{tot}}_t = \sum_{i=1}^{N}(p_{i,t} + p^c_{i,t}) + p^c_{s,t}$$

SCENARIO



PROBLEM FORMULATION

$$\min_{\boldsymbol{\Psi}_t} \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{p^{\text{tot}}_\tau\}$$

subject to
$(a)\ \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{L_\tau\} \le \overline{\mathrm{L}}; \to$ Avg. Latency Constraint

$(b)\ \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{G_\tau\} \ge \overline{\mathrm{G}}; \to$ Avg. Learning Performance Constraint

$(c)\ \lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\{\alpha_\tau\} = \overline{\alpha}; \to$ Avg. Convergence Rate Constraint

$$\left.\begin{array}{l} b_{i,t} \in \mathcal{C}_i,\ \forall i \in \mathcal{S}_t, t; \quad R^{\min}_i \le R_{i,t} \le R^{\max}_{i,t},\ \forall i \in \mathcal{S}_t,\ t; \\ f^{\min}_i \le f_{i,t} \le f^{\max}_i,\ \forall i \in \mathcal{S}_t,\ t; \quad v_{k,l,t} \in \mathcal{R},\ \forall k,l,t; \\ B_t \in \mathcal{B},\ \forall t; \quad f^{s,\min} \le f^s_t \le f^{s,\max},\ \forall t; \end{array}\right\} \mathcal{X}_t$$

where $\boldsymbol{\Psi}_t = [\boldsymbol{v}_t, \{b_{i,t}\}_{i\in\mathcal{S}_t}, \{R_{i,t}\}_{i\in\mathcal{S}_t}, \{f_{i,t}\}_{i\in\mathcal{S}_t}, f^s_t, B_t]$

ONLINE LEARNING METRICS

- Generally no closed-form expression for G_t and α_t
- Online estimation in a totally data-driven fashion
- $\widehat{G}_t$ and $\widehat{\alpha}_t$ to estimate online G_t and α_t, e.g. for classification:

$$\widehat{G}_t = \frac{1}{|\mathcal{T}|}\sum_{y\in\mathcal{T}}\mathbb{I}(\widehat{y}_t = y), \quad \widehat{\alpha}_t = \frac{1}{\kappa}\sum_{\tau=t-\kappa}^{\kappa-1}(\widehat{G}_\tau - \widehat{G}_{\tau-1})$$

LYAPUNOV STOCHASTIC OPTIMIZATION

- Virtual Queues:
  - Z_t for the Latency inequality constraint:

$$Z_{t+1} = \max\left\{0, Z_t + \epsilon_z\left(L_t - \overline{\mathrm{L}}\right)\right\} \tag{1}$$

  - Q_t for the accuracy inequality constraint

$$Q_{t+1} = \max\left\{0, Q_t + \epsilon_q\left(\overline{\mathrm{G}} - \widehat{G}_t\right)\right\} \tag{2}$$

  - Y_t for the convergence rate equality constraint:

$$Y_{t+1} = [Y_t + \epsilon_{y,t}(\widehat{\alpha}_t - \overline{\alpha})] \cdot \mathbb{I}\left(\widehat{G}_t \le \overline{\mathrm{G}}\right) \tag{3}$$

- Drift-plus-penalty function:

$$\Delta^p_t = \mathbb{E}\Big\{\frac{1}{2}(Z^2_{t+1} + Q^2_{t+1} + Y^2_{t+1}) - \frac{1}{2}(Z^2_t + Q^2_t + Y^2_t) + V \cdot p^{\text{tot}}_t \Big| \boldsymbol{\Phi_t}\Big\},$$

where $\boldsymbol{\Phi_t} = [Z_t, Q_t, Y_t]$.

ALGORITHMIC SOLUTION

- ***Step 1:*** $\forall t$, observe $\boldsymbol{\Phi_t}$ and minimize a DPP upper bound instantaneous values [2]:

$$\min_{\boldsymbol{\Psi}_t\in\mathcal{X}_t} \quad Z_t\widetilde{L}_t - Q_t\widetilde{G}_t - Y_t\widetilde{\alpha}_t + V \cdot p^{\text{tot}}_t \tag{4}$$

Mixed-integer non linear optimization problem, closed-form solutions for any given $\mathcal{S}_t$, $\{b_{i,t}\}_{i\in\mathcal{S}_t}$, B_t and $\boldsymbol{v}_t$, $\to$ Find $\mathcal{S}_t$, $\{b_{i,t}\}_{i\in\mathcal{S}_t}$ and $\boldsymbol{v}_t$ with the proposed two-stage greedy selection, setting:

$$R_{i,t} = \left[\frac{2B_i}{\ln(2)}\,W\!\left(\frac{\ln(2)}{B_i}\sqrt{\frac{Z_t\, m \cdot b_{i,t}\, h_{i,t}(\boldsymbol{v}_t)}{2VN_0}}\right)\right]_{R^{\min}_i}^{R^{\max}_{i,t}} \tag{5}$$

$$f_{i,t} = \left[\left(\frac{Z_t B_t J_i}{3\gamma_i V}\right)^{\frac{1}{4}}\right]_{f^{\min}_i}^{f^{\max}_i} \qquad f^r_t = \left[\left(\frac{Z_t C\,|\mathcal{S}_t|}{3\gamma_s V}\right)^{\frac{1}{4}}\right]_{f^{r,\min}}^{f^{r,\max}} \tag{6}$$

- ***Step 2:*** Update Z_t as (1), Q_t as in (2), Y_t as in (3).

GREEDY SELECTION OF $\{v_{k,t}\}_{k=1}^{K}$

- The method greedily selects $\{\boldsymbol{v}_{k,t}\}_{k=1}^{K}$ to maximize:

$$\Delta^R(\{\boldsymbol{v}_{k,t}\}_{k=1}^{K}) = \sum_{i=1}^{N}\delta_{i,t}\left|h^a_{i,t} + \sum_{k=1}^{K}\boldsymbol{h}^T_{i,k,t}\,\mathrm{diag}(\boldsymbol{v}_{k,t})\,\boldsymbol{z}^a_{i,k,t}\right|^2$$

where $\delta_{i,t} = \dfrac{1/|h^a_{i,t}|^2}{\sum_{i=1}^{N} 1/|h^a_{i,t}|^2}$ assigns more importance to devices that experience worse instantaneous channel conditions without RISs

- Polynomial complexity in K, M and $|\mathcal{R}|$

GREEDY SELECTION OF $\mathcal{S}_t$

- For each $B_t \in \mathcal{B}$, the method starts from $\mathcal{S}_t = \emptyset$ and iteratively adds the most convenient devices, selecting $\{b_{i,t}\}_{i=1}^{N}$ and the edge resources as in (5) and (6)
- The method keeps adding devices until the value of the objective in (4) decreases
- Polynomial complexity in N, $\max_i\{|\mathcal{C}_i|\}$, $|\mathcal{B}|$

REFERENCES

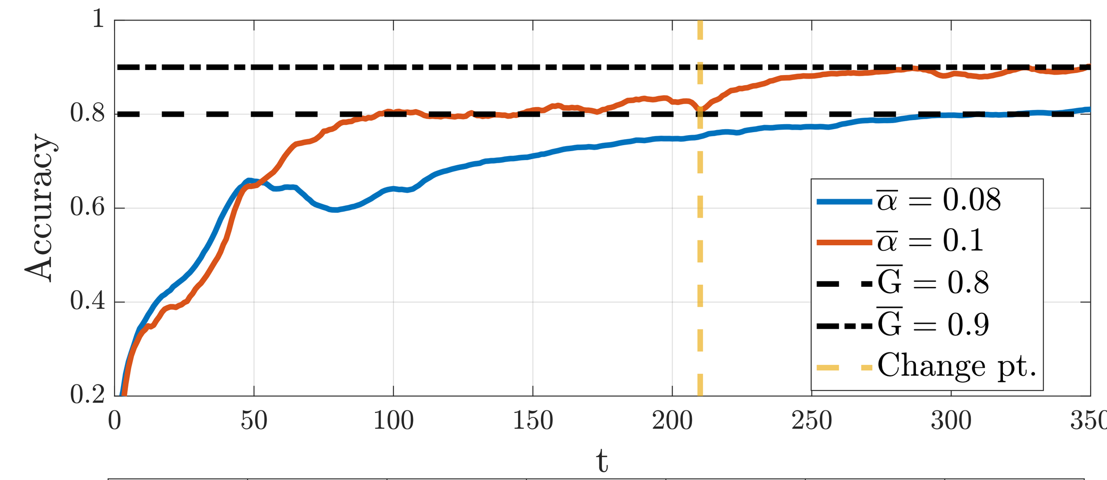
[1] T. D. Burd and R. W. Brodersen. Processor design for portable systems. *J. VLSI Signal Process. Syst.*, 13(2-3):203–221, Aug. 1996.

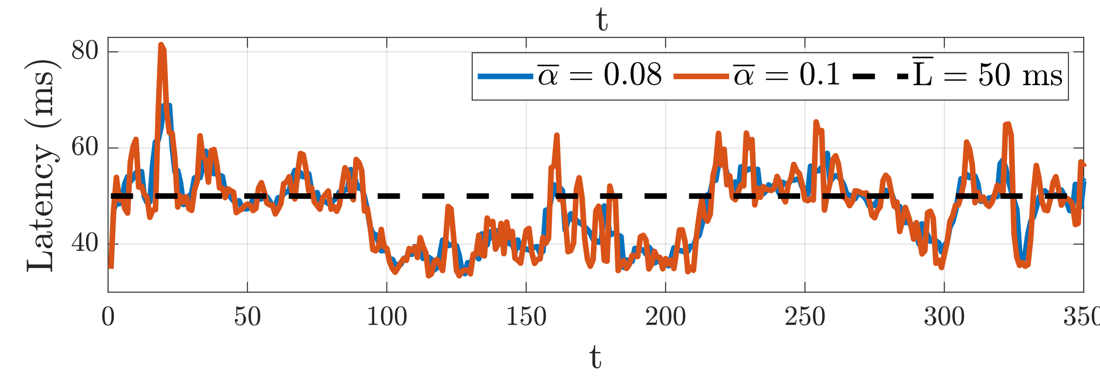
[2] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Morgan and Claypool Publishers, 2010.

SIMULATION SET UP

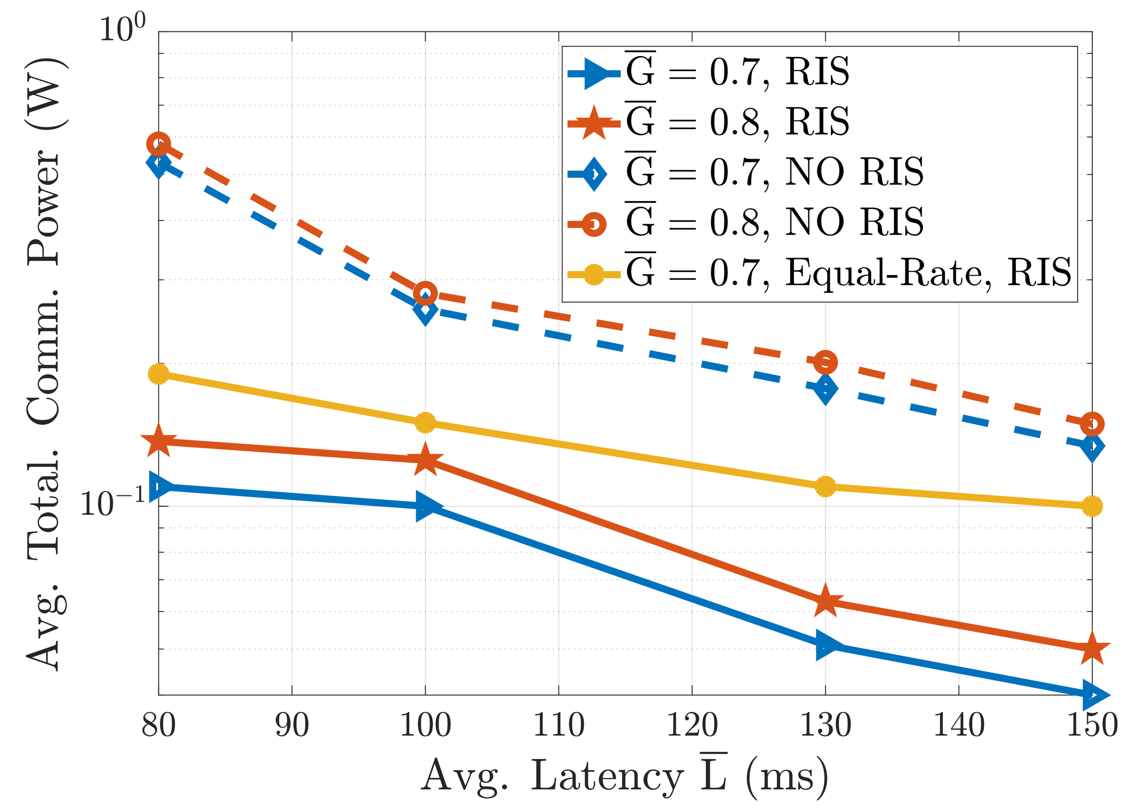
- $N = 9$ devices and one AP equipped with an edge server
- Classification task on the MNIST dataset (10 classes)
- CNN with 4 convolutional layers (~100K parameters)
- ADAM Optimizer, learning rate 0.001, forgetting factors $\beta_1 = 0.9$, and $\beta_2 = 0.99$
- One RIS equipped with 1-bit discrete phase shifters
- The channels are generated using the ABG model, using a carrier frequency equal to 6 GHz, with a unit variance Rayleigh fading

NUMERICAL RESULTS





- The method guarantees the prescribed performance in terms of $\overline{\alpha}$ and $\overline{\mathrm{G}}$, within $\overline{\mathrm{L}}$
- The method reacts promptly to changes in the accuracy requirement



- Baseline given by an equal-rate policy with all the agents always transmitting
- The tradeoff gets worse imposing a stricter $\overline{\mathrm{G}}$ requirement
- Significant gain obtained thanks to the presence of the RIS

CONCLUSIONS

- We proposed an online strategy for adaptive federated learning empowered by reconfigurable intelligent surfaces (RISs)
- The strategy dynamically minimizes the power expenditure of the system, while guaranteeing target learning performance and latency constraints in a fully data driven fashion
- The strategy allows the exploration of a new trade-off of communication networks, including power expenditure, delay, and learning performance
- Numerical results on federated training of Deep Neural Networks illustrate the advantages obtained by the proposed strategy and by the usage of RISs