IEEE International Conference on Communications Workshops, ICC (2020)

Energy-Efficient Radio Resource Allocation for Federated Edge Learning

Authors:
Qunsong Zeng
Yuqing Du
Kaibin Huang
Kin K. Leung†

Presented By:
Ittehad Saleh Chowdhury
Reyadath Ullah

Federated Edge Learning (FEEL)

FEEL, or **Federated Edge Learning**, is a distributed machine learning paradigm that enables edge devices to collaboratively learn a shared model while keeping the data localized, thus improving privacy and reducing communication overhead.

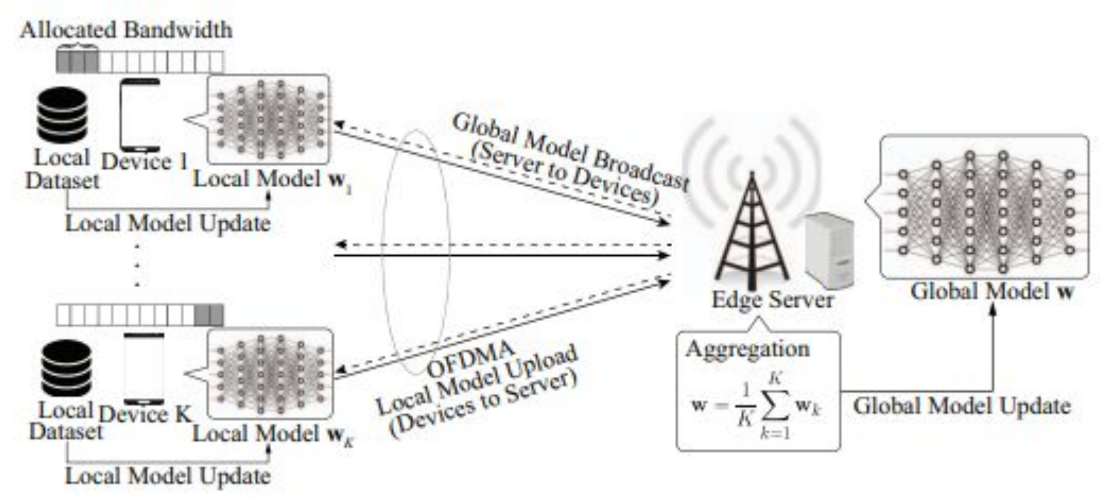


Figure 1. A framework for FEEL system.

Energy Efficient Radio Resource Management (RRM) for FEEL

Radio Resource Management (**RRM**) for **FEEL** refers to the strategies and algorithms used to allocate and manage radio spectrum resources—like bandwidth and transmitting power—among multiple edge devices to facilitate efficient and reliable communication for distributed machine learning tasks.

Energy-Efficient **RRM** within **FEEL** specifically aims to:

- Minimize Energy Consumption
- Balance Trade-offs
- Adapt to Device Capabilities and Network Conditions

Main challenges in Federated Edge Learning

- Communication Overhead
- Device Heterogeneity
- Latency
- Energy Constraints
- Privacy and Security

Objectives of this Paper

Common Focus in FEEL Research:

Most research in the field prioritizes optimizing wireless communication to quicken the learning process by minimizing communication overhead and latency in Federated Edge Learning.

Main Objective of This Paper:

- Minimize energy consumption of edge devices.
- Prolong battery life of edge devices through optimized energy consumption.
- Introduce RRM approaches that prioritize energy efficiency without compromising learning performance.

Multiple Access Model

In this model **Orthogonal Frequency-Division Multiple Access(OFDMA)** which is a multi-user version of the digital modulation scheme that divides a communication channel into multiple orthogonal sub-channels for simultaneous data transmission by several users is considered.

Allocated Bandwidth (for device k) = $\gamma_k B$

$\gamma_k \in [0,1]$ = Bandwidth allocation ratio, B = Total Bandwidth

Time Constraint :

$$t_k^{\text{comp}} + t_k \leq T, \quad \forall k \in \mathcal{K}, \qquad (1)$$

t_k^{comp} = Model training time,

t_k = uploading time, T = Maximum total time for one Communication Round

Energy Consumption Model

The achievable rate denoted by r_k is give by Shannon Capacity formula:

$$r_k = \gamma_k B \log\left(1 + \frac{p_k h_k^2}{N_0}\right), \qquad (2)$$

p_k = Transmission Power(Watt/Hz)

h_k = Channel gain

N_o = variance of channel noise

The rate can also be defined in terms of the data size(L):

$$r_k = \frac{\beta_k L}{t_k}, \qquad (3)$$

β_K = 1 if device k is selected for uploading, or 0 otherwise

Energy Consumption Model (Cont.)

By combining (2) and (3) the **Uploading** energy consumption can be determined:

$$E_k^{\mathrm{up}} = \gamma_k B p_k t_k = \frac{\gamma_k B t_k N_0}{h_k^2}\left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right). \qquad (4)$$

E^{comp} which is the Energy Consumption for Local Training is considered the same for all devices

Learning Speed Model

The convergence of stochastic gradient descent (SGD) can be accelerated by involving more devices for global updating. Hence, the total number of participating devices can be a measure of the learning rate.

$$\text{(Learning speed)} \quad \sum_{k=1}^{K} \beta_k.$$

Energy Efficient Bandwidth Allocation

The goal is to minimize the total energy consumption i.e. $\sum_{k=1}^{K}\left(E^{\text{comp}} + E_k^{\text{up}}\right)$

Since, E^{comp} is uniform and fixed the problem focuses on minimizing the uploading time.

$$
\begin{aligned}
(\mathbf{P1}) \quad & \min_{\{\gamma_k, t_k\}} \ \sum_{k=1}^{K} \frac{\gamma_k B t_k N_0}{h_k^2}\left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1\right) \\
& \text{s.t.} \ \sum_{k=1}^{K} \gamma_k = 1, \ \ 0 \le \gamma_k \le 1, \ k \in \mathcal{K}, \\
& \qquad 0 \le t_k \le T_k, \ k \in \mathcal{K}.
\end{aligned}
$$

By solving the above problem, the server can optimally determine the bandwidth partitioning, as specified by $\{\gamma_k\}$, and the uploading time $\{t_k\}$ for the devices.

Optimal Bandwidth Allocation

The optimal policy for bandwidth allocation is:

$$\gamma_k^\star = \frac{\beta_k L \ln 2}{BT_k \left[1 + \mathcal{W}\left(\frac{h_k^2 \nu^\star - BT_k N_0}{BT_k N_0 e}\right)\right]}, \ k \in \mathcal{K}, \qquad (6)$$

$$t_k^\star = T_k, \ k \in \mathcal{K}, \qquad (7)$$

Corollary: $\gamma_k^\star$ is a non-increasing function with respect to T_k and h^2_k, respectively. Therefore,

- More bandwidths should be allocated to devices with weaker computation capacities, namely smaller T_k
- More bandwidths should be allocated to devices with weaker channels

ENERGY-AND-LEARNING AWARE SCHEDULING

By modifying Problem **(P1)** to include the learning speed in the objective, the current problem can be formulated as:

$$
\begin{aligned}
\mathbf{(P2)} \quad \min_{\{\gamma_k, t_k, \beta_k\}} \quad & \sum_{k=1}^{K} \frac{\gamma_k B t_k N_0}{h_k^2} \left(2^{\frac{\beta_k L}{\gamma_k B t_k}} - 1 \right) - \lambda \sum_{k=1}^{K} \beta_k \\
\text{s.t.} \quad & \beta_k \in \{0, 1\}, \; k \in \mathcal{K}, \\
& \sum_{k=1}^{K} \gamma_k = 1, \quad 0 \le \gamma_k \le 1, \; k \in \mathcal{K}, \\
& 0 \le t_k \le T_k, \; k \in \mathcal{K},
\end{aligned}
$$

where the trade-off factor $\lambda > 0$ is a predetermined constant.

Continued

- λ determines the trade-off between minimizing energy consumption and maximizing device participation in the learning process.

- Adjusting λ alters the optimization focus, either reducing total power usage or increasing the number of active learning devices.

Solving P2

- To solve this problem, we adopt the common method of relaxation-and rounding.
- It firstly relaxes the integer constraint **$\beta k \in \{0, 1\}$** as as **$0 \le \beta k \le 1$**
- Two sub-problems have been proposed
    - The bandwidth-allocation in Problem **(P1).**
    - User scheduling problem.

The first problem is allocating bandwidth given scheduled devices indicated by **$\{\beta k\}$.** The other subproblem (user scheduling) is to decide on the selection priorities of the devices.

Edge-device Selection Priority

$$\beta_k^\star = \min\left\{\max\left\{\frac{\gamma_k B T_k}{L}\log\left(\frac{\lambda h_k^2}{N_0 L \ln 2}\right), 0\right\} 1\right\}, \;\; k \in \mathcal{K}. \tag{8}$$

Algorithm 1

Algorithm 1 Joint Bandwidth Allocation and User Scheduling

Initialization: Randomly set indicators $\{\beta_k\} \in [0, 1]$.

Iteration:

- **(Energy-efficient Bandwidth Allocation)**: Given fixed $\{\beta_k\}$, compute $\{\gamma_k, t_k\}$ using (6) and (7);
- **(Energy-and-Learning Aware Scheduling)**: Given fixed $\{\gamma_k, t_k\}$, compute $\{\beta_k\}$ using (8);

Until Convergence.

Round indicators $\{\beta_k\}$ to $\{0, 1\}$.

Compute $\{\gamma_k, t_k\}$ using (6) and (7).

Output the optimal solution $\{\beta_k^\star, \gamma_k^\star, t_k^\star\}$.

IMPROVEMENT BY OPPORTUNISTIC SPECTRUM ACCESS

- Bandwidth allocation is initially fixed but can dynamically adjust based on real-time device capabilities and conditions.

- Faster devices can opportunistically access unused spectrums, identified via spectrum sensing, for more energy-efficient uploading.

- The process involves initial scheduling with Algorithm 1, followed by dynamic spectrum utilization as per Algorithm 2, enhancing efficiency.

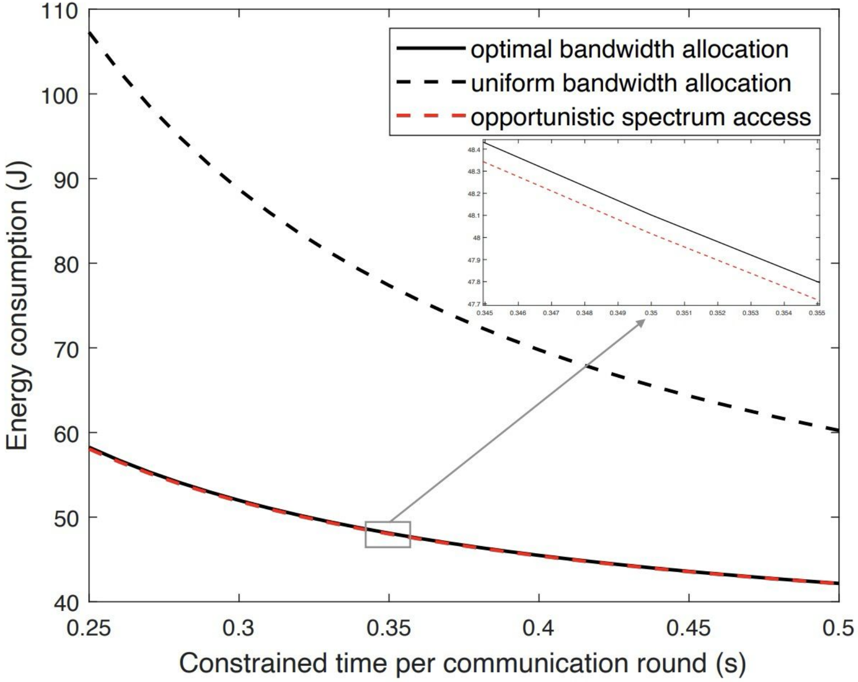
Algorithm 2

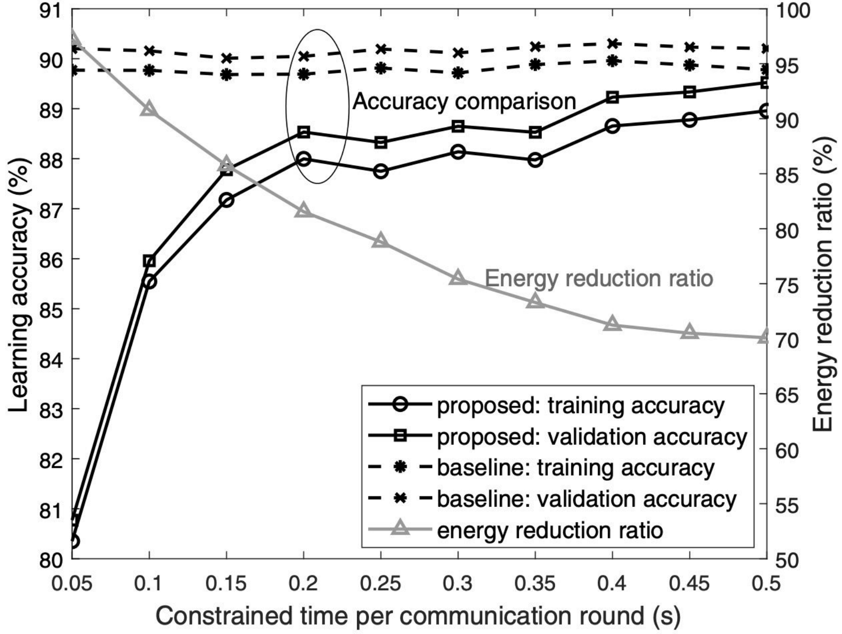
Algorithm 2 Opportunistic Spectrum Access

Initialization: Apply Algorithm 1 to obtain $\{\beta_k^\star, \gamma_k^\star\}$.
Denote τ as the time slot duration and let $t_{\mathsf{count}} = 0$. For the subset of devices $\mathcal{S} = \{k \in \mathcal{K} \mid \beta_k^\star = 1\}$:
While $t_{\mathsf{count}} < T$:

- Denote $\mathcal{S}_l$ the set of devices that have not completed local computation at time t_{count}. For $k \in \mathcal{S}_l$, no bandwidth will be occupied by them;
- $\forall k \in \mathcal{S}/\mathcal{S}_l$, besides the allocated bandwidth $\gamma_k^\star B$, each device could sense additional $\frac{\sum_{k \in \mathcal{S}_l} \gamma_k^\star B}{|\mathcal{S}/\mathcal{S}_l|}$ bandwidths;
- $t_{\mathsf{count}} = t_{\mathsf{count}} + \tau$;

Simulation Results





Potential Future Improvements

- Extend energy-efficient RRM strategies to asynchronous model-update scenarios.

- Develop more detailed models of energy consumption for local computing.

- Adapt strategies to better handle heterogeneous network device capabilities.

Thank You!