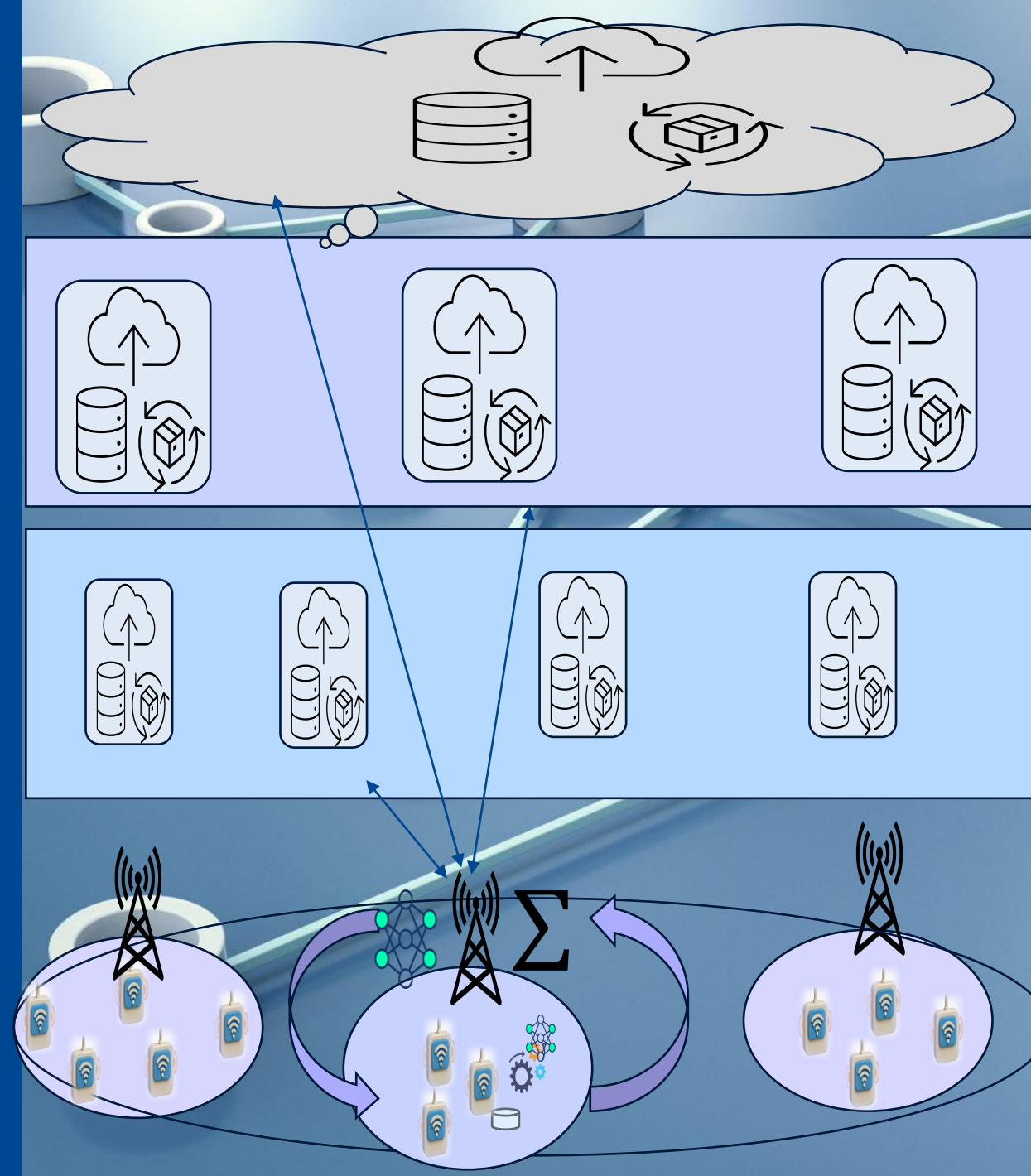


# Task-Centric Sustainable Federated Learning over the Edge–Cloud Continuum

Swapnil Sadashiv Shinde  
NSE Department, KTH





# About Me

- **Swapnil Sadashiv Shinde**  
Postdoctoral Researcher, KTH  
Supervisor: Prof. Carlo Fischione
- **Research Focus:** Machine Learning for Wireless Communication
- **Background:** PhD in Automotive for Intelligent Mobility, University of Bologna, Italy
- **Current Work:** Advancing research in intelligent wireless networks

## Today's Presentation is based on these works

- a) Swapnil Sadashiv Shinde, Krishnendu Sankunni Tharakan, and Carlo Fischione, "*Hierarchical Multi-Objective Deep Reinforcement Learning for Task-Centric Federated Learning over the Edge-Cloud Continuum.*" (Under Preparation)
- b) Swapnil Sadashiv Shinde, David Naseh, Daniele Tarchi, and Carlo Fischione, "*LEO Satellites Accelerating Edge Intelligence: Model Transfer and Similarity-Aware Initialization for Federated Learning*", IEEE Internet of Things Magazine, (Major Revision, March-2026)
- c) Swapnil Sadashiv Shinde, and Carlo Fischione, "*Task-Centric Model Transfer for Federated Learning over the Edge-Cloud Continuum*", Presented in IEEE MECOM 2025, Cairo, Egypt



# Outline

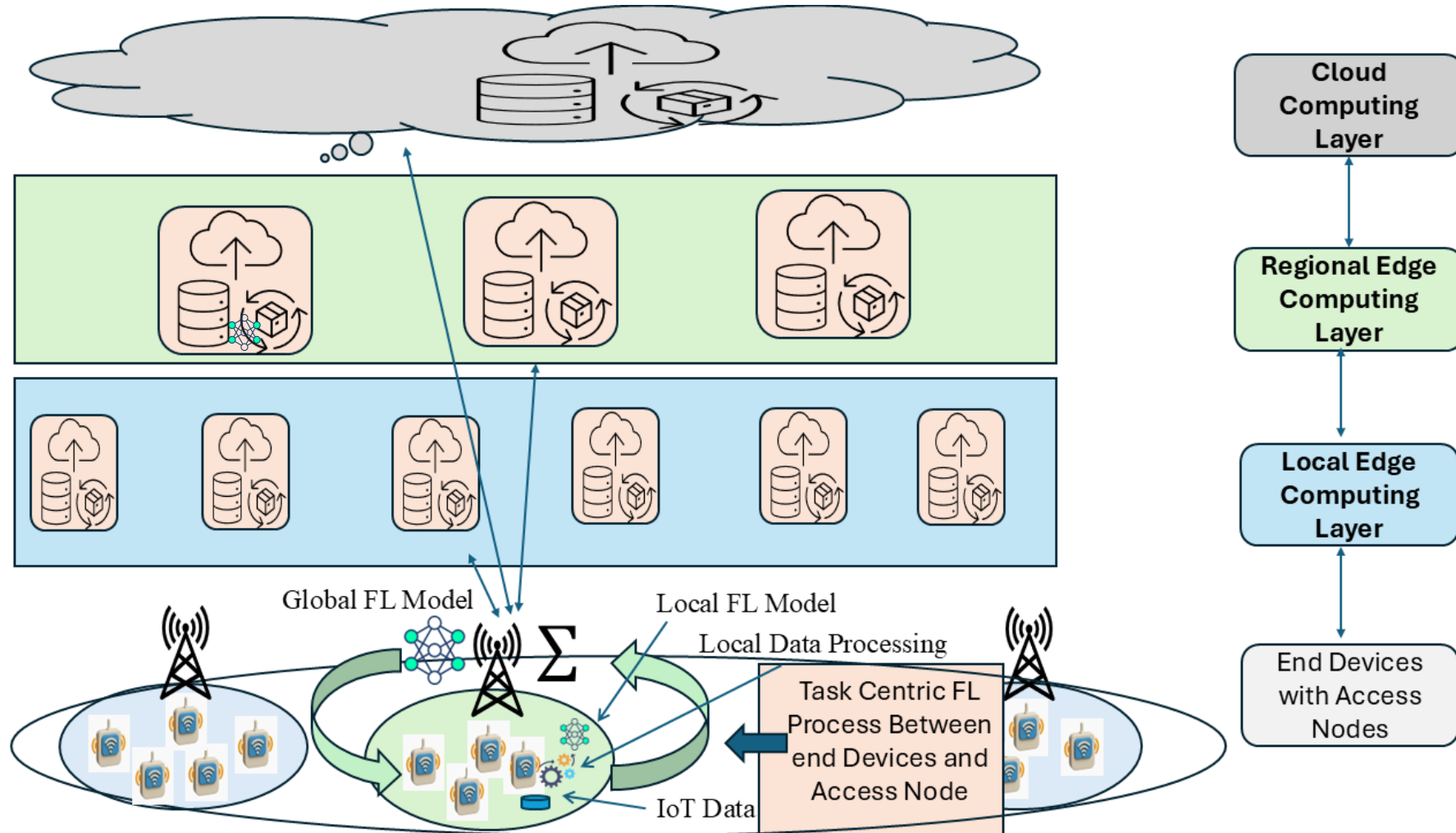
- Motivation and our contributions
- Introduction
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL)
- Performance evaluation
- Conclusion and future directions



# Outline

- **Motivation and our contributions**
- Introduction
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL)
- Performance evaluation
- Conclusion and future directions

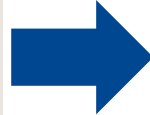
# Federated Learning over the Edge-Cloud Continuum



# Motivation and Our Contributions (I)

## Remote IoT Devices with Task-Specific Learning Needs

- Limited computation & energy
- High energy cost for communication & training
- Need sustainable and low-latency intelligence



## Edge–Cloud Continuum

- Distributed computing (edge → cloud)
- Heterogeneous resources & coverage

+

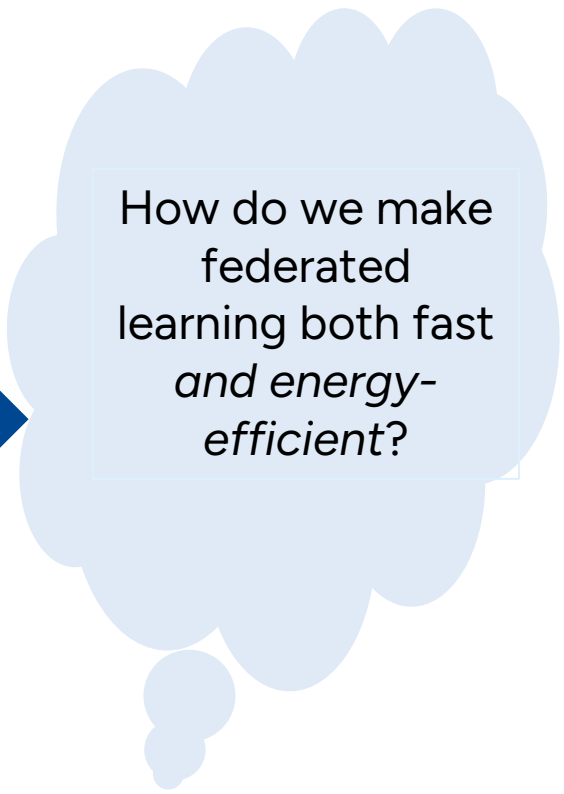
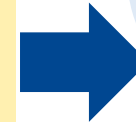
## Federated Learning (FL)

- Privacy-preserving distributed learning
- High cost due to iterative training

+

## Model Reuse / Transfer Learning

- Reuse pre-trained models across regions
- Potential to reduce training time



How do we make federated learning both fast *and* energy-efficient?

# Motivation and Our Contributions (II)

## Our Contributions

- **Task-Centric FL with Model Reuse**
  - Reuse pre-trained models across edge–cloud layers
- **Joint Latency–Energy Optimization**
  - Balances search, transfer, and training
- **Hierarchical Multi-Objective RL Framework**
  - Learns adaptive model selection policies
  - Enables efficient placement & decision-making

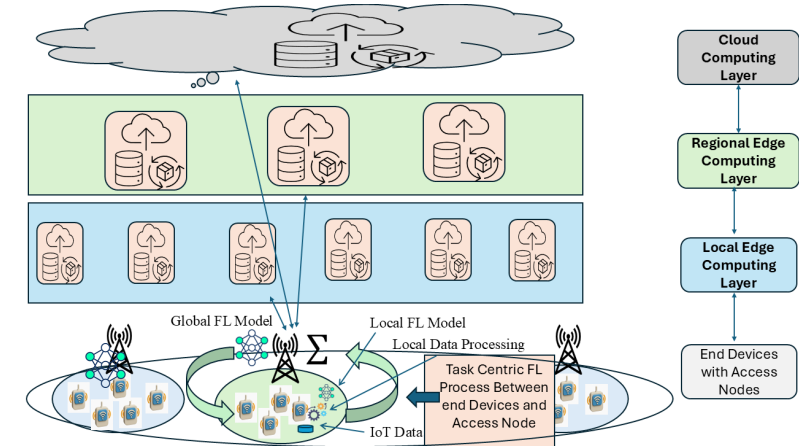
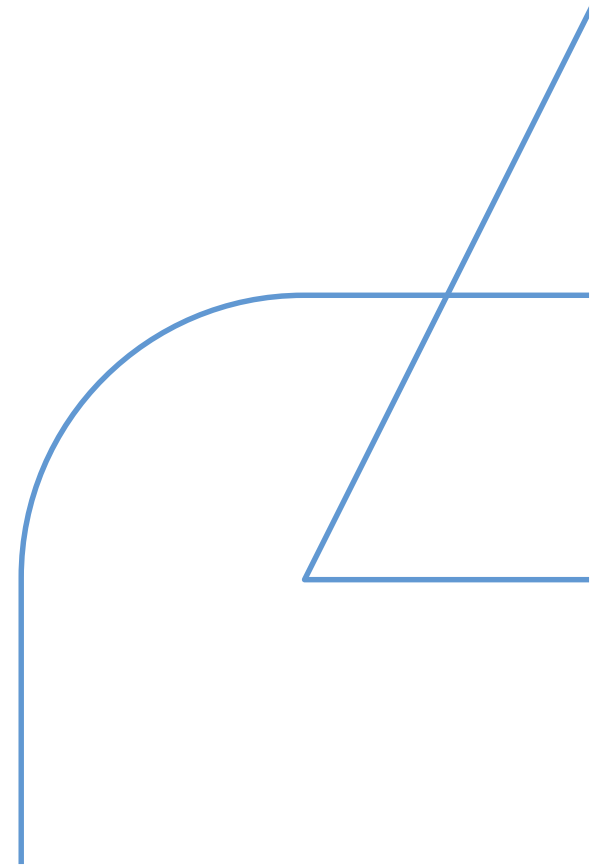


Fig: Task Centric FL with Model Transfer



# Outline

- Motivation and Our Contributions
- **Introduction**
  - Task Centric FL for IoT Applications
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL)
- Performance evaluation
- Conclusion and future directions



# Introduction: Task Centric IoT-FL

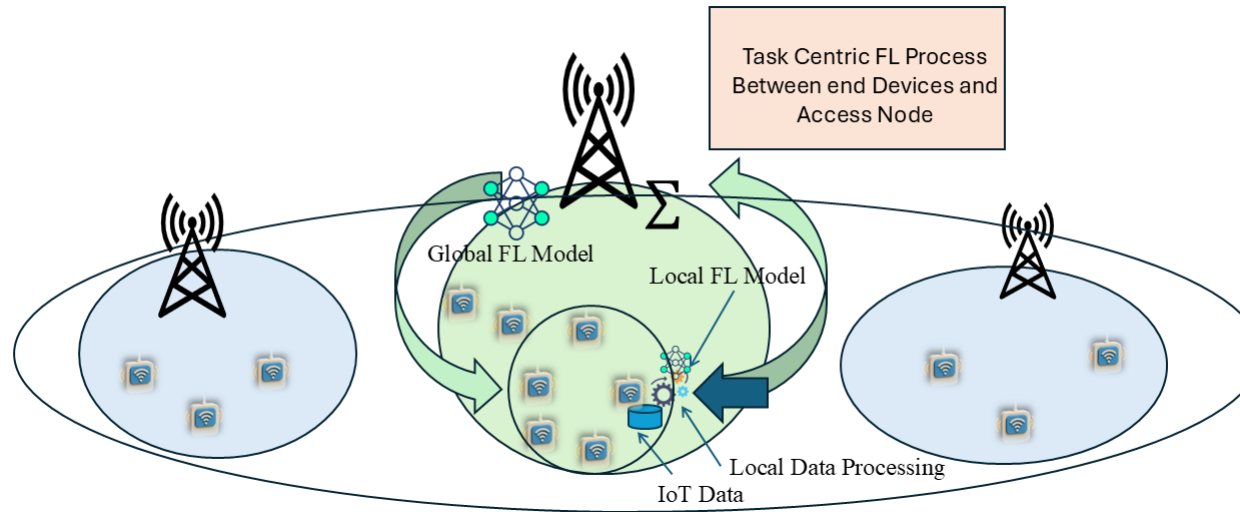


Fig: Task Centric IoT-FL

## ■ Task Centric IoT-FL

- Task-driven learning across IoT devices
- Learning is driven by specific application tasks (not generic models)
- Selective participation of devices based on task relevance



# Outline

- Motivation and our contributions
- Introduction
- **System Model and Problem Formulation**
  - System Model
  - Cost Analysis
  - Joint Latency and Energy Minimization
- Hierarchical Multi-Objective RL (H-MORL)
- Performance evaluation
- Conclusion and future directions

# System Model and Problem Formulation: System Model

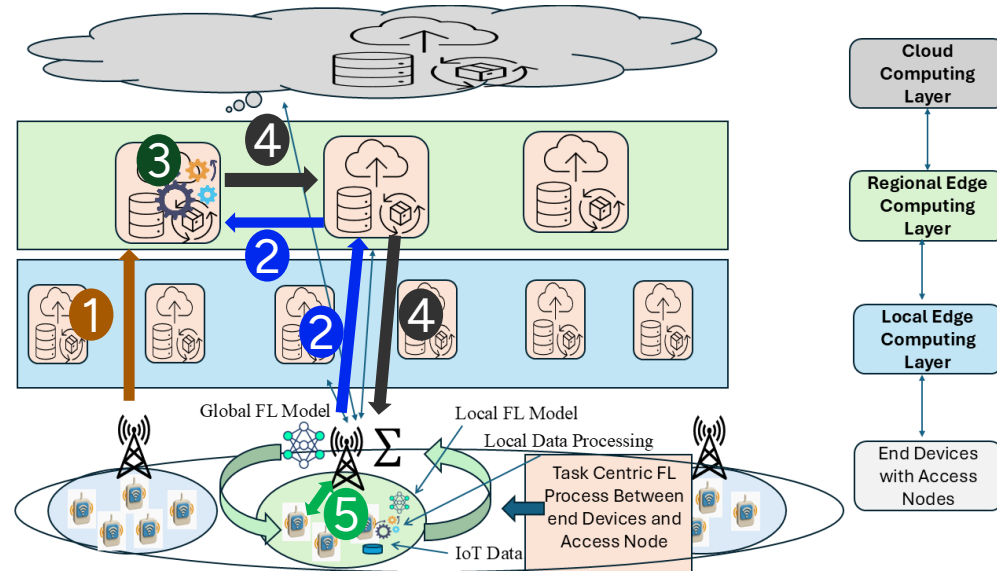


Fig: Task Centric FL with Model Transfer

- **Hierarchical Edge–Cloud Architecture**  
→ local edge, regional edge, and cloud layers with heterogeneous resources
- **Task-Centric Federated Learning Setup**  
→ IoT devices collaboratively train models via an access node using local data
- **Distributed Model Repository with Transfer Capability**  
→ Pre-trained models stored across layers can be selected and reused based on task similarity

How to optimally select an edge–cloud layer and server for appropriate model selection?

# System Model and Problem Formulation: Cost Analysis

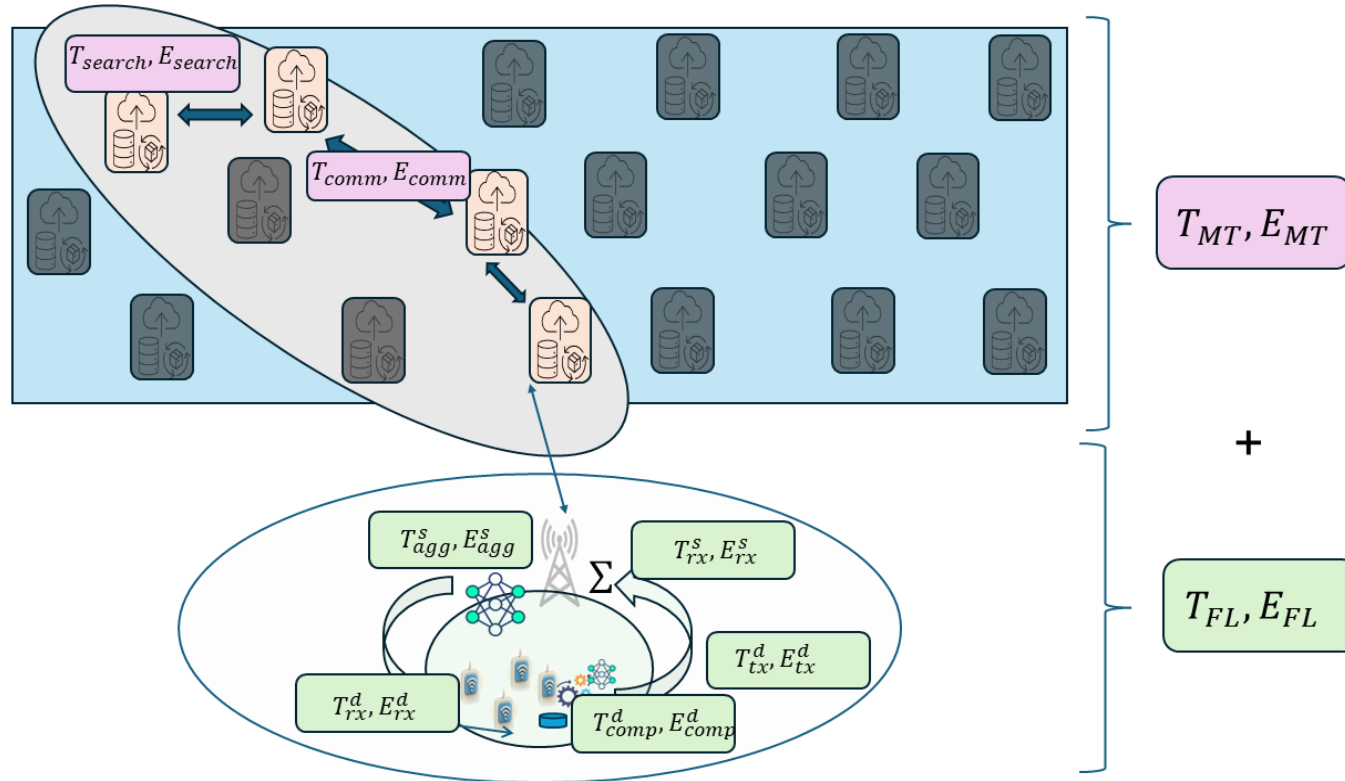


Fig: Latency and Energy Flow Diagram

## ■ Total Cost Decomposition

- Model Transfer Cost ( $T_{MT}, E_{MT}$ )
- Federated Learning Cost ( $T_{FL}, E_{FL}$ )
- **Key trade-off: better initialization vs added overhead**



# System Model and Problem Formulation: Joint Latency and Energy Minimization

- **Objective Function:** Joint Latency and Energy Minimization

- minimize:

$$\alpha \cdot T_{total} + (1 - \alpha) \cdot E_{total}$$

- where:

$$T_{total} = T_{MT} + T_{FL}$$

$$E_{total} = E_{MT} + E_{FL}$$

- **Constraints**

- C1: Resource constraint
- C2: Model similarity
- C3: FL iterations bound
- C4: Server selection

## ■ Motivation

- Problem is inherently multi-objective
- Large combinatorial decision space (layer + server selection)
- Heterogeneous environment
- Conventional methods struggle to scale
- Need an intelligent, adaptive



# Outline

- Motivation and our contributions
- Introduction
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL) MDP Formulation
  - H-MORL Approach
  - H-MORL Framework
- Performance evaluation
- Conclusion and future directions

# Hierarchical Multi-Objective RL (H-MORL) (I)

## High-Level Decision (Layer Selection)

### Where to search?

- Edge (low latency, limited models)
- Regional (balanced trade-off)
- Cloud (rich models, higher delay)

## Low-Level Decision (Server Selection)

### Which server to select?

- Evaluate model similarity
- Consider transmission cost
- Select best candidate model

## Multi-Objective Optimization

### Minimize:

- Search cost
- Transmission latency
- FL training cost (latency & energy)

## Key Insight

- MO-HRL enables scalable decision-making by decomposing the problem and optimizing multiple conflicting objectives

# Hierarchical Multi-Objective RL (H-MORL) (II)

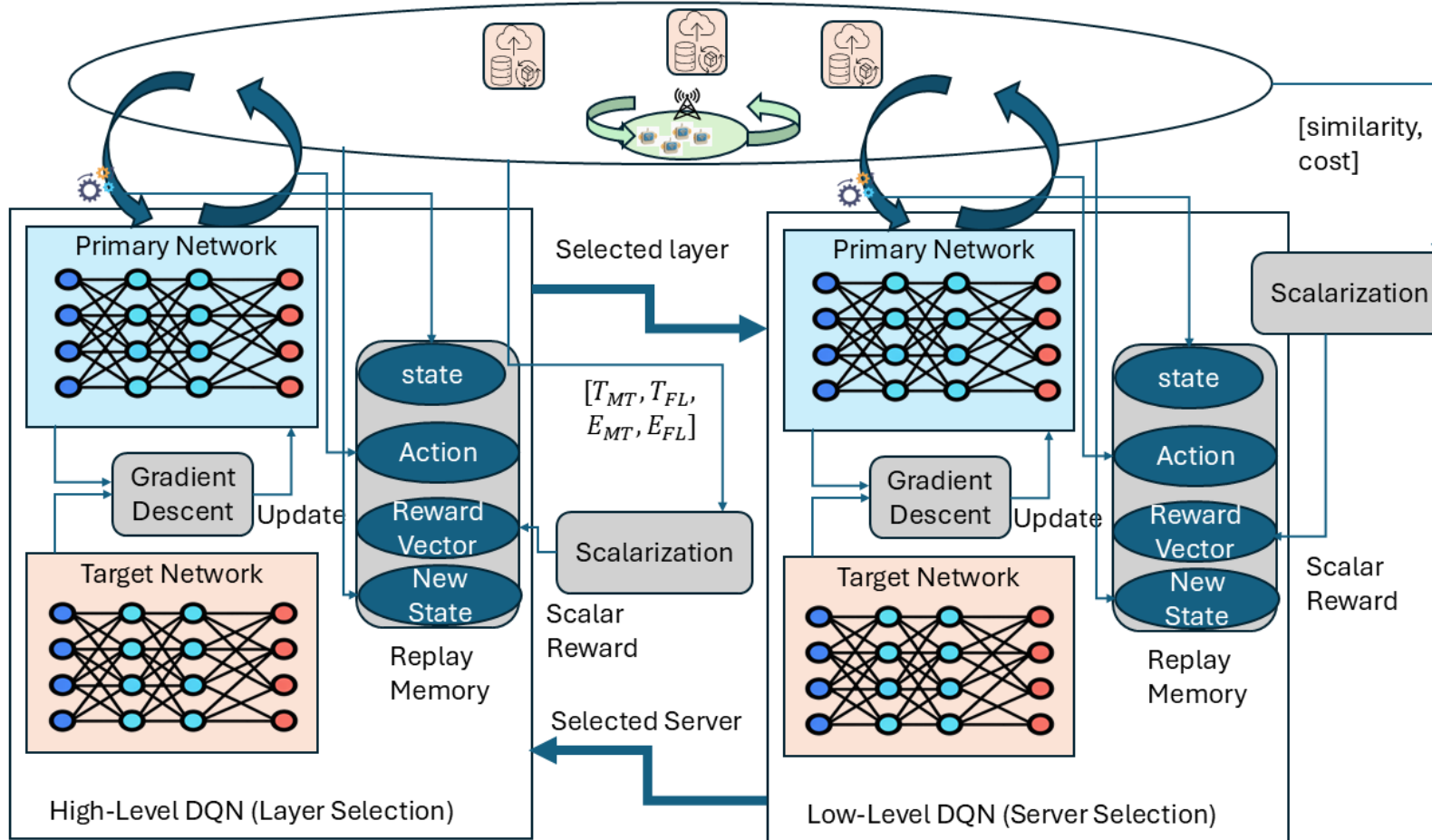


Fig: H-MORL Framework for Joint Layer and Server Selection in Edge-Cloud FL



# Outline

- Motivation and our contributions
- Introduction
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL)
- **Performance evaluation**
  - Benchmark Solutions
  - Simulation Parameters
  - Cost Analysis
- Conclusion and future directions

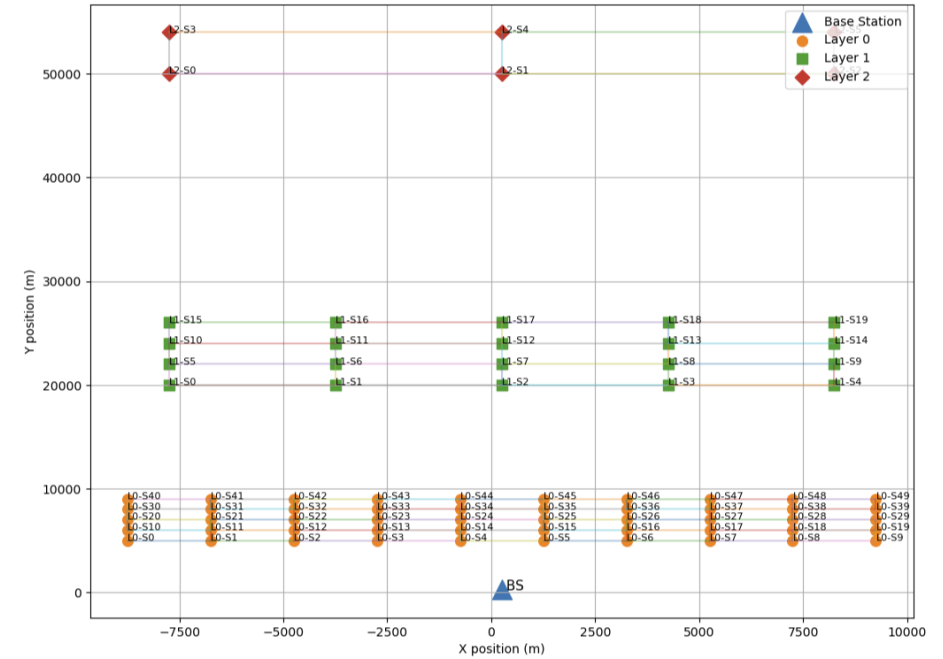


# Performance Evaluation: Benchmark Solutions

- **Traditional FL**
  - Zero model search and transmission cost
  - Improper utilization of edge resources
- **FL with MT (Random Layer and Node Selection Approach)**
  - Unpredicted performance

# Performance Evaluation: Simulation Parameters

- Network Topology:
  - 3-layer hierarchical edge network
  - IoT Users: 50
- Learning Parameters
  - Discount factor:  $\gamma = 0.95$
  - Learning rate:  $10^{-3}$
  - Batch size: 32
  - Replay buffer: 5000
  - Target update frequency: 20
  - Episodes: ~200



```
# Predefined FL parameters
FL_Devices = [10,20,30,40,50,60,70,80,90,100]
FL_Task_Type = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]
FL_Model_Type = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20]
FL_Data_Type = [1,2,3]
FL_Device_Samples = [100,200,300,400,500,600,700,800,900,1000]

# All possible models
all_models = list(itertools.product(
    FL_Devices,
    FL_Task_Type,
    FL_Model_Type,
    FL_Data_Type,
    FL_Device_Samples
))
all_models_array = np.array(all_models)
```

# Performance Evaluation: Training Performance

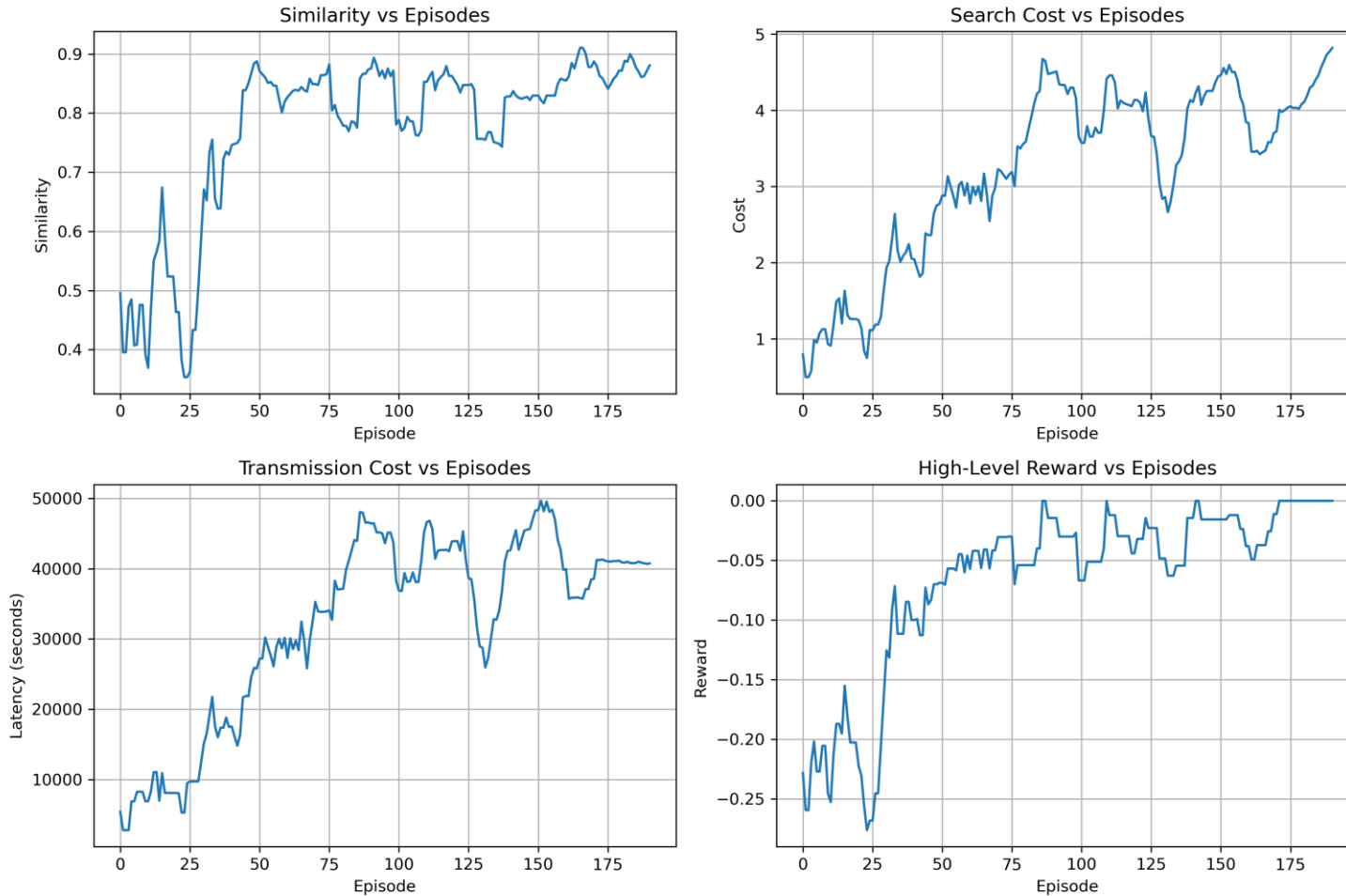
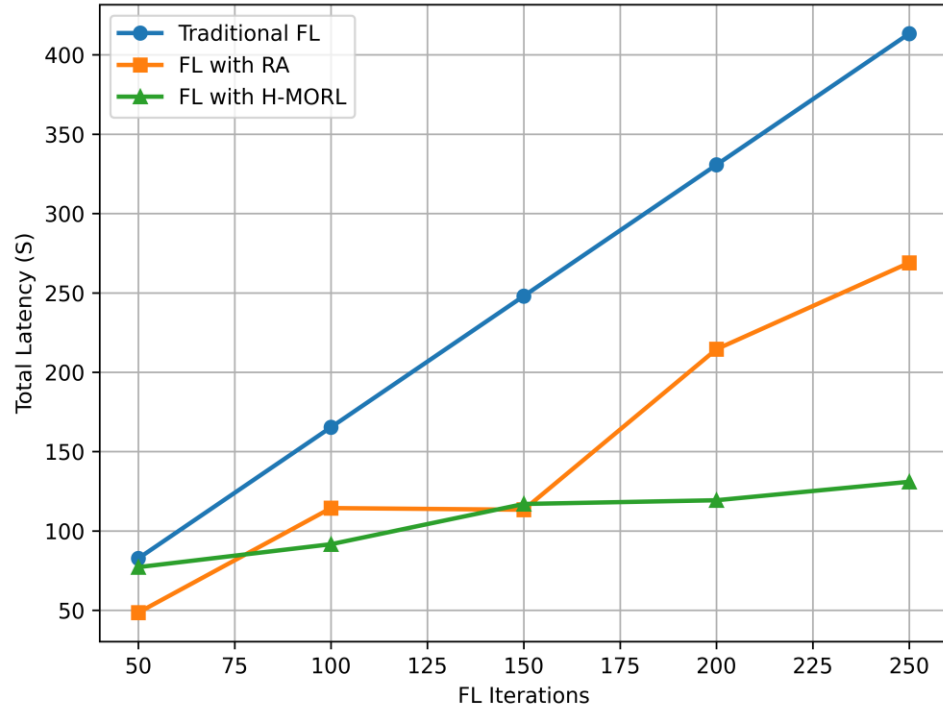


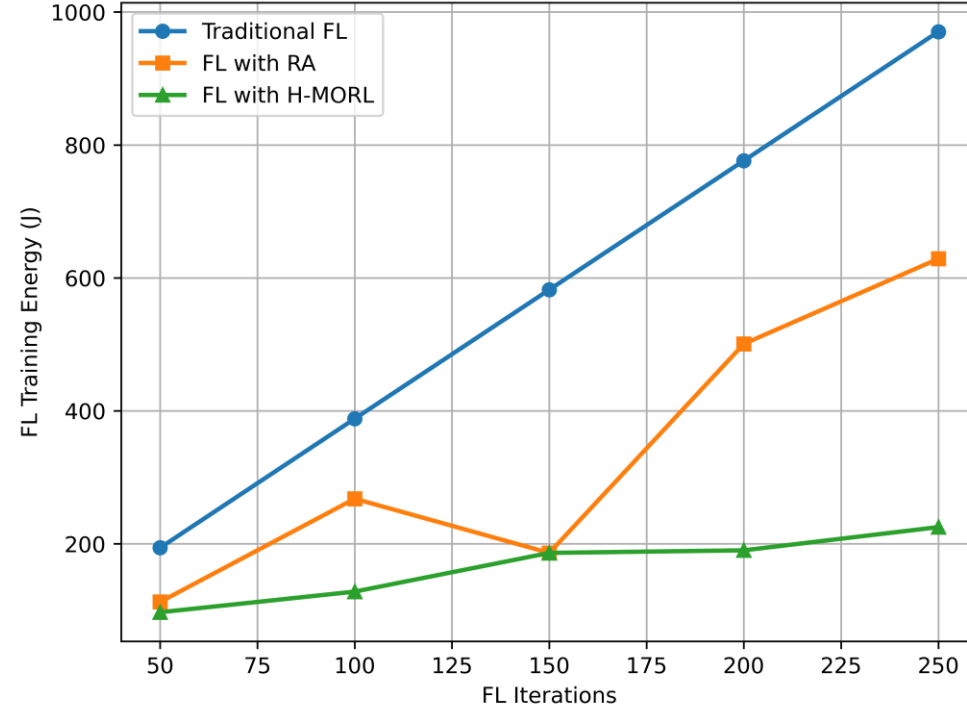
Fig.: H-MORL Training Performance

- We observe general trends of increasing similarity and improving reward, while the costs show some stabilization over training, indicating learning progress.

# Performance Evaluation: Cost Analysis (I)



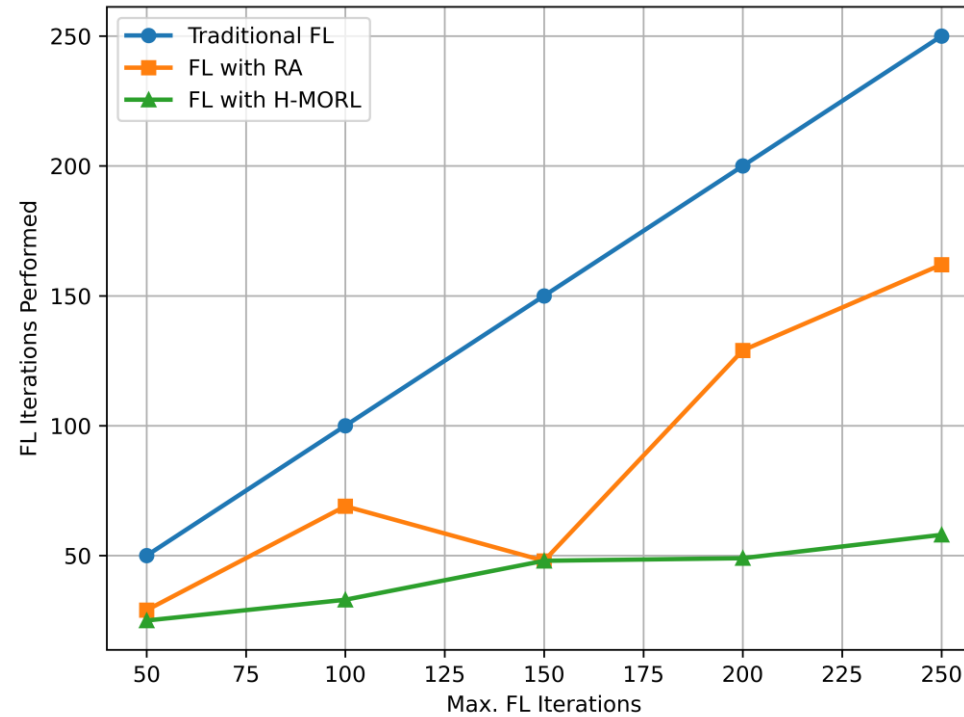
**Fig.: Total Latency Cost**



**Fig.: Total Energy Cost**

- H-MORL significantly reduces total latency compared to baselines
- Energy consumption is consistently lower with H-MORL
- Traditional FL shows highest cost due to training from scratch
- Random selection (RA) improves performance but remains suboptimal

# Performance Evaluation: Cost Analysis (II)



**Fig.: FL Iterations Performed**

- With pre-trained model search, H-MORL requires significantly fewer FL iterations



# Outline

- Motivation and our contributions
- Introduction
- System Model and Problem Formulation
- Hierarchical Multi-Objective RL (H-MORL)
- Performance evaluation
- **Conclusion and future directions**



# Key Takeaways

- **Task-centric Federated Learning with Model Transfer:** Leveraging pre-trained models across the edge–cloud continuum reduces training time and improves energy efficiency.
- **Trade-off Between Model Search/Transmission Cost and Training Speed:** Finding better (more similar) models reduces FL iterations, but introduces additional search and transmission latency.
- **Hierarchical Multi-Objective RL Enables Optimal Decisions:** The proposed H-MORL framework intelligently balances these trade-offs, minimizing overall latency in distributed FL systems.



# Conclusions and Future Directions

## ■ Conclusions

- Proposed task-centric federated learning with model transfer
- Developed an intelligent H-MORL strategy for layer and server selection
- Jointly optimized latency and energy in edge–cloud systems
- Achieved faster convergence with fewer FL iterations
- Significantly reduced overall system cost

## ■ Future Directions

- Scalable deployment in large-scale real-world IoT systems
- Incorporating dynamic network and mobility conditions
- Extending to multi-task and continual learning scenarios
- Integration with satellite/LEO-assisted edge intelligence



KTH

VETENSKAP  
OCH KONST