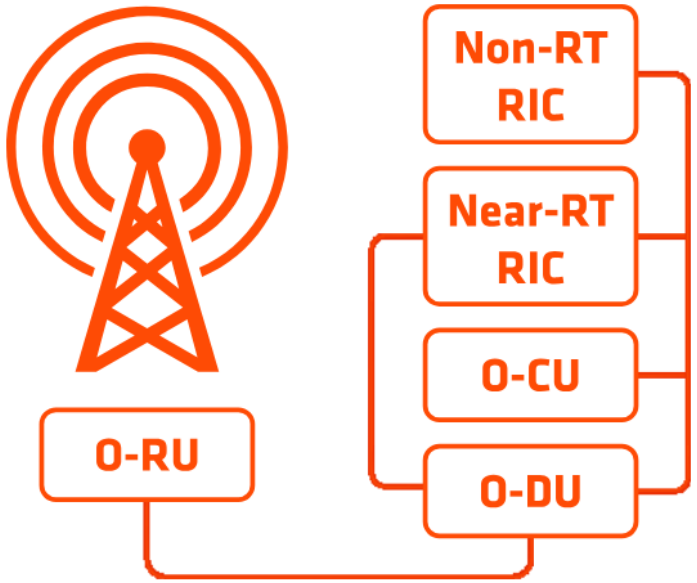
Energy Saving for Cell-Free Massive MIMO Networks: A Multi-Agent Deep Reinforcement Learning Approach

Presenter: Qichen Wang

Cell-Free Massive MIMO in O-RAN Architecture

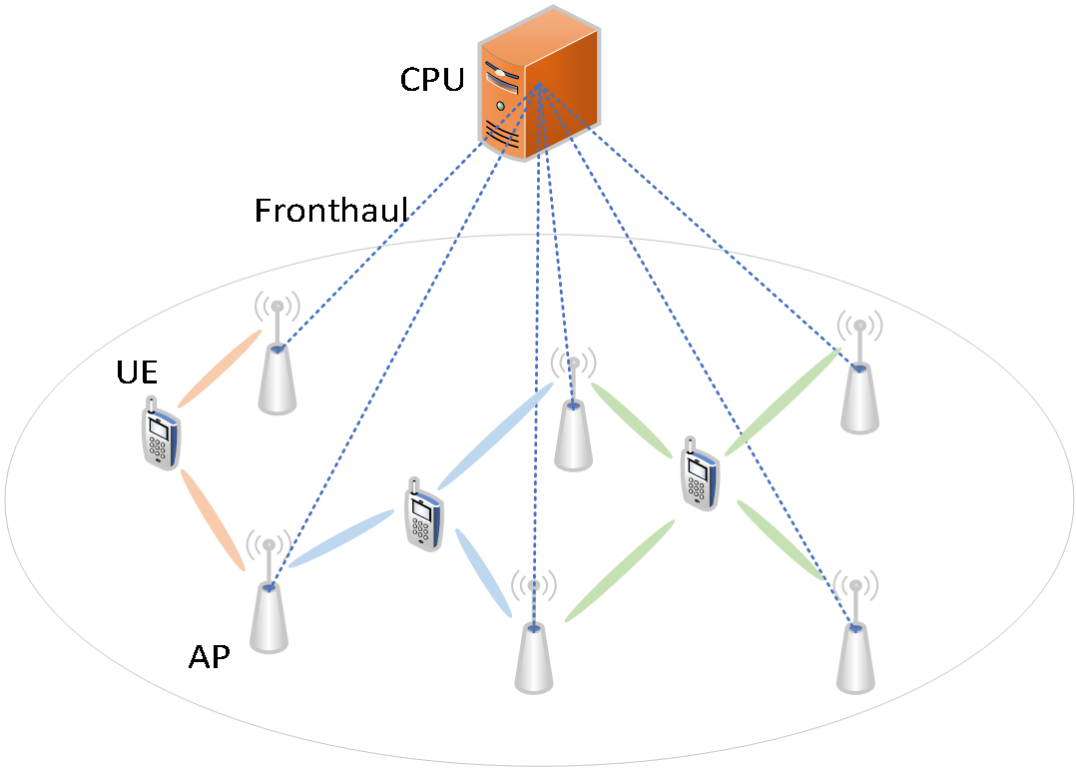
O-RAN Components:

- ***Non-RT RIC:*** long-term policy training, data-driven optimization
- ***Near-RT RIC:*** short-term control and coordination
- ***O-CU / O-DU / O-RU:*** handle protocol layers and radio operations



Cell-Free Massive MIMO:

- Each UE is jointly served by a subset of distributed APs
- CPU coordinates APs via fronthaul links
- Enables uniform service quality across the entire coverage area



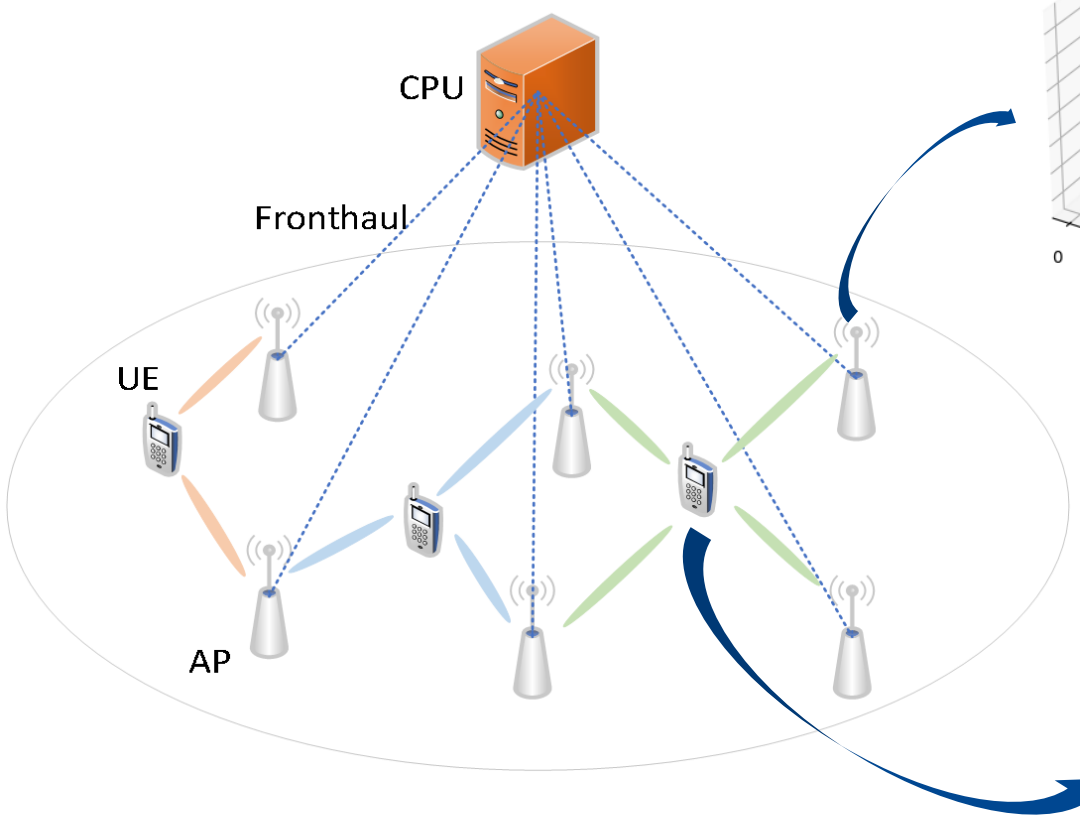
Source: Firecell, “Learn Open RAN,” https://firecell.io/learn/open-ran/.

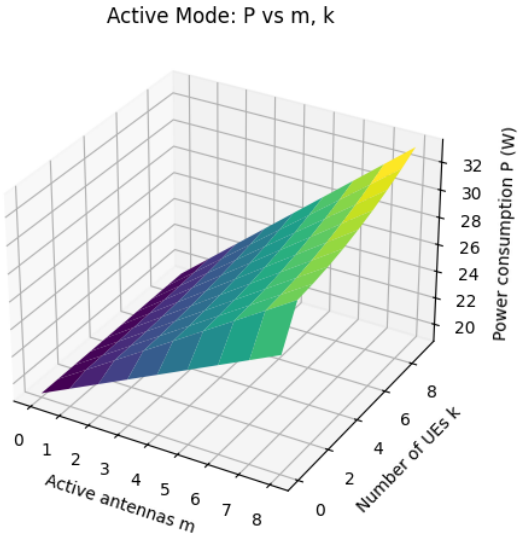
Why AI?

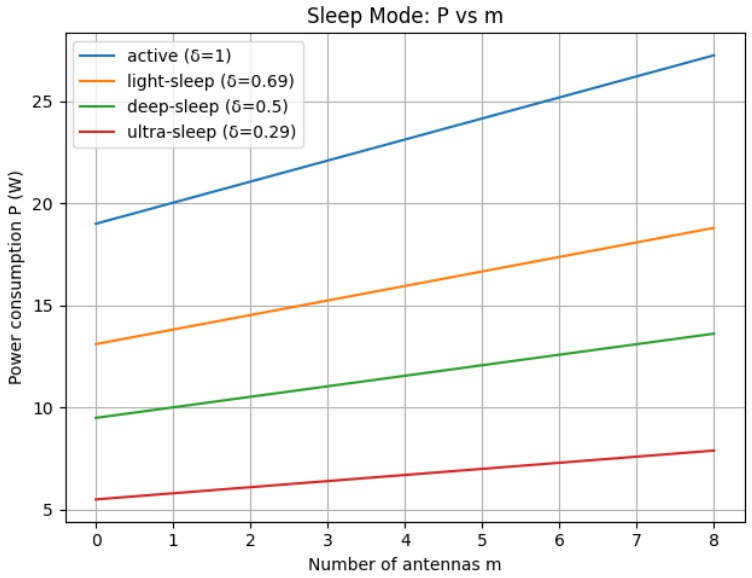
Aspect	Traditional Optimization	AI-based Approach
Non-convexity handling	Requires convexification or approximations; often yields local optima.	Learns complex nonlinear relationships and achieves near-optimal solutions without convexity assumptions.
Real-time adaptability	Iterative solving is slow; unsuitable for dynamic network conditions.	Enables fast, real-time decision making through trained models.
Model dependency	Depends on accurate analytical models and full system knowledge.	Data-driven; effective even with imperfect or unknown models.
Generalization ability	Needs re-optimization under new traffic or environmental changes.	Learns transferable policies that generalize to unseen scenarios.

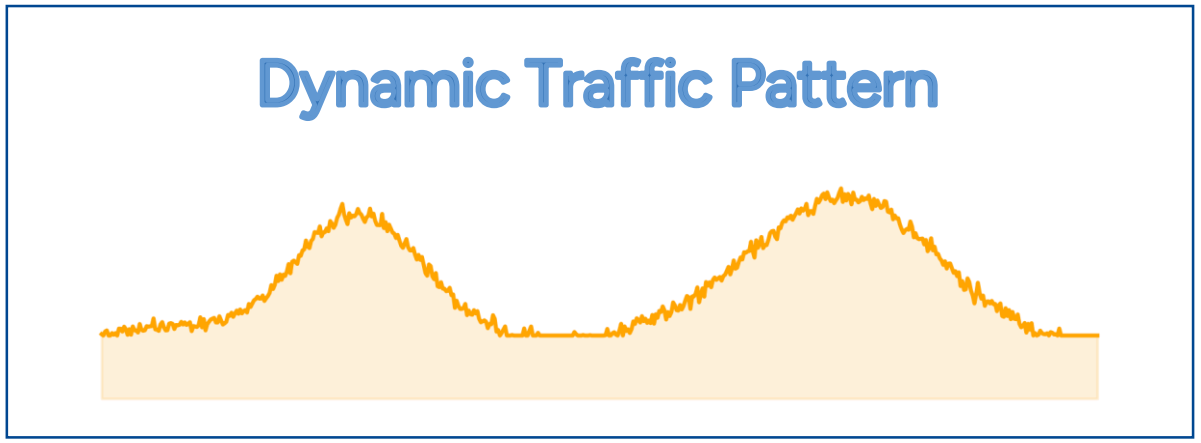
- AI-based methods overcome the limitations of traditional optimization by enabling data-driven, real-time, and adaptive energy-efficient control in dynamic networks.

System Model

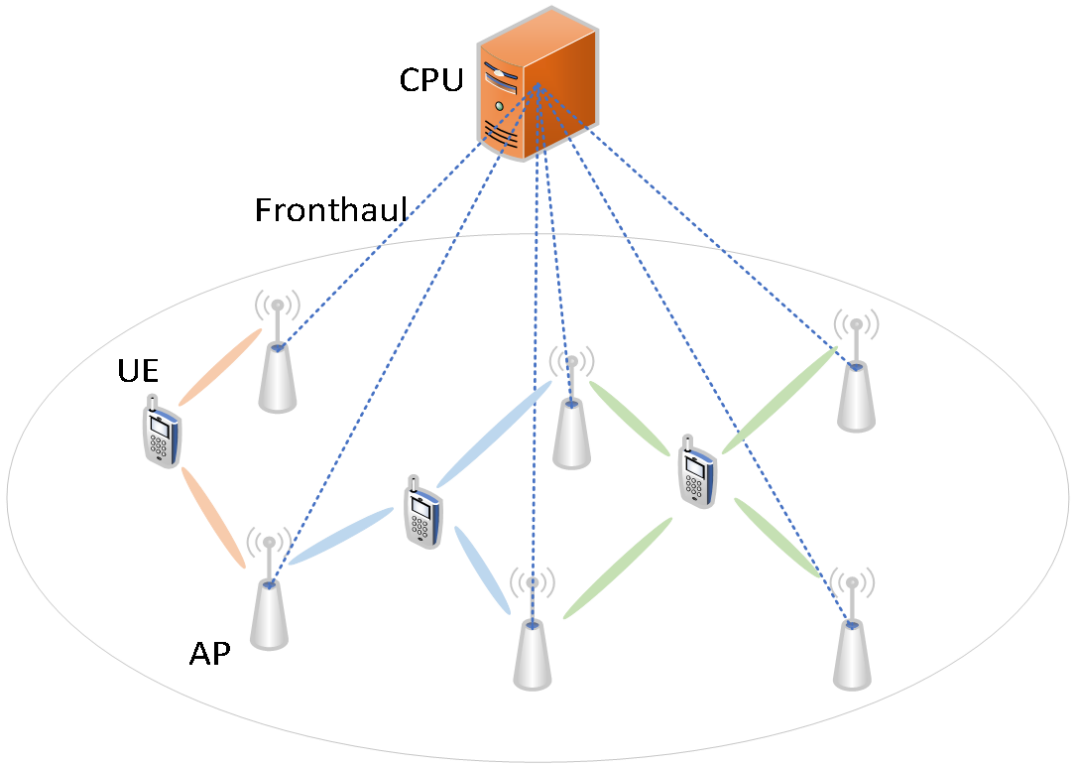








Problem Formulation



$$\min_{s_l, m_l, \forall l} \quad P_{\text{net}}$$

$$\text{s.t.} \quad \frac{1}{K}\sum_{k=1}^{K} \rho_k \geq 1 - \delta_{\text{drop}}, \qquad \forall k,$$

$$m_l \in \{0, 1, \ldots, M_{\max}\}, \qquad \forall l,$$

$$\tau_l^{\text{str}} = \begin{cases} 0, & m_l = 0, \\ \leq m_l - 1, & m_l \geq 1, \end{cases} \qquad \forall l,$$

$$s_l \in \{0, 1, 2, 3\}, \qquad \forall l.$$

Goal:

- Minimize power consumption P_{net}

Optimization:

- Antenna activation m_l
- Sleep mode s_l

Constraints:

- Drop ratio (QoS) δ_{drop}
- Pilot signals τ_l^{str}

RL Algorithm Design

State S:

Each agent observes partial local information (e.g., PC, number of activated antennas, sleep mode) and aggregate UE statistics (e.g., total demand and achieved rate), along with neighboring AP states.

Action A:

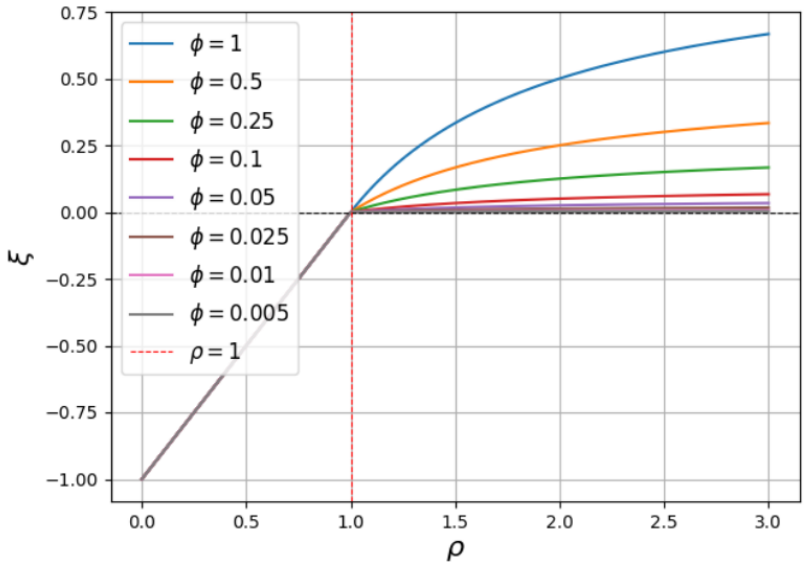
$A = A_m \times A_s$

- $A_m = \{-1, 0, 1\}$: antenna switching
- $A_s = \{0, 1, 2, 3\}$: sleep level

Reward R:

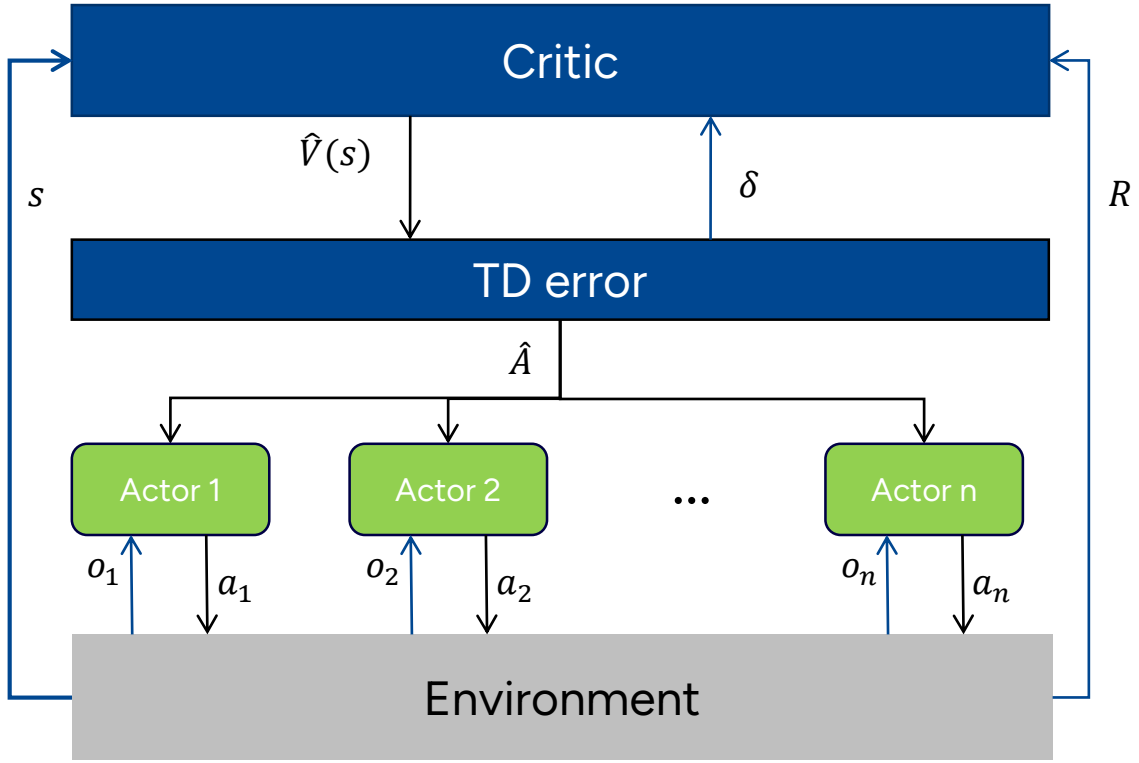
$$R = w_{\text{rs}} \frac{1}{K} \sum_{k=1}^{K} \xi_k - w_{\text{pc}} P_{\text{net}},$$

- Rate satisfaction: $\xi_k = \begin{cases} \rho_k - 1, & \rho_k < 1, \\ \phi\left(1 - \frac{1}{\rho_k}\right), & \rho_k \geq 1. \end{cases}$
- ρ_k: $\dfrac{\text{achieved rate of UE } k}{\text{demanded rate of UE } k}$



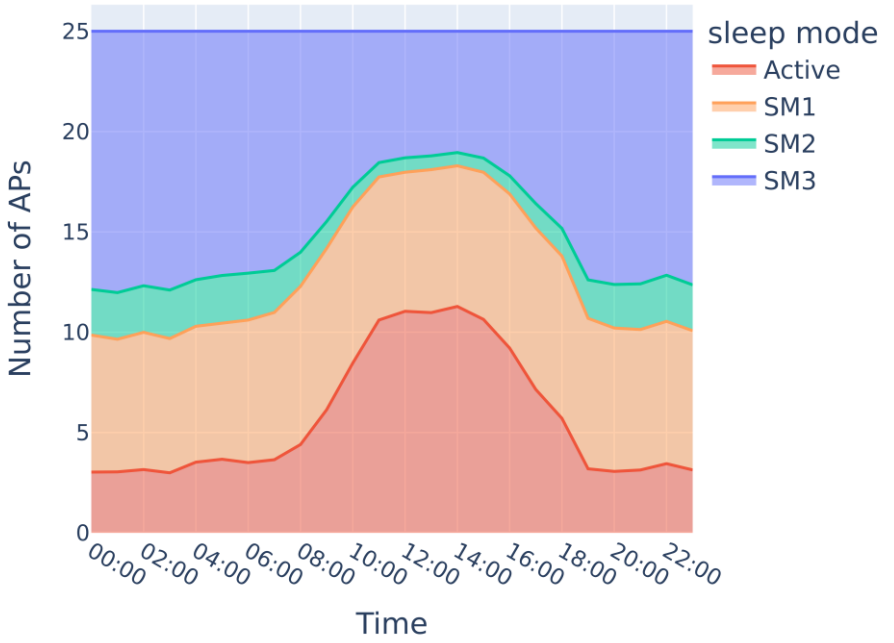
Multi Agent Proximal Policy Optimization (MAPPO)

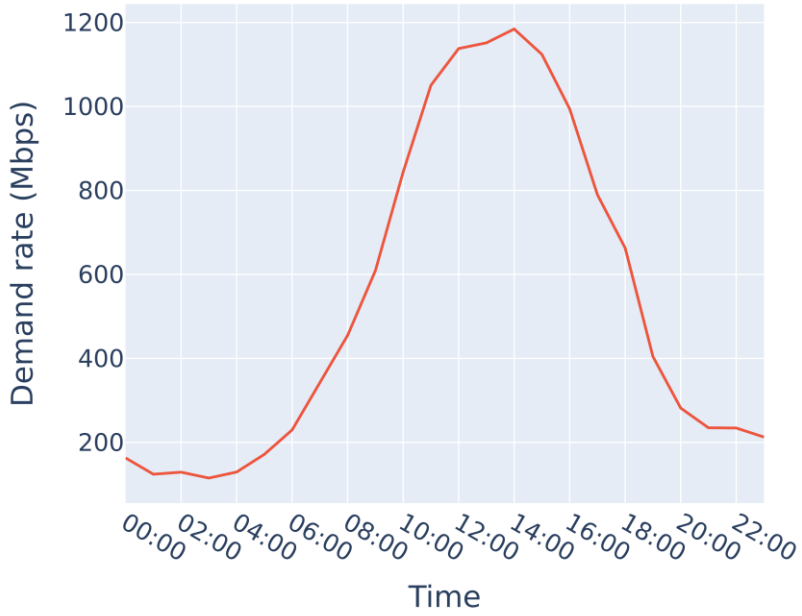
- ***Independent actor:*** independent policy
- ***Centralized critic:*** stationary learning signal
- ***Global reward:*** optimizing a shared objective

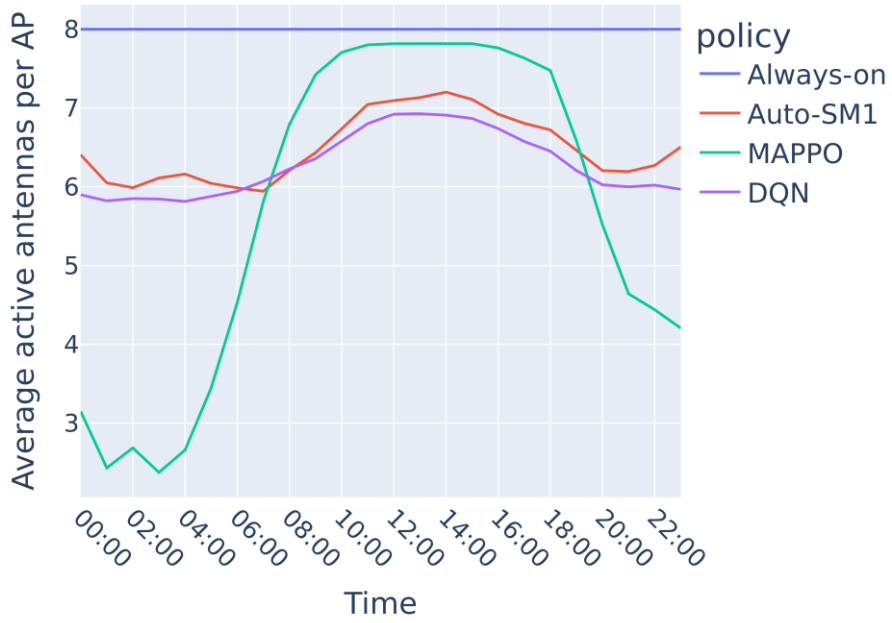


Simulation Results

- ***Always-on:*** vanilla policy
- ***Auto-SM1:*** load-dependent mechanism
- ***DQN:*** widely used RL algorithm
- ***MAPPO***-based algorithm can dynamically adapt antenna and sleep according to traffic condition.

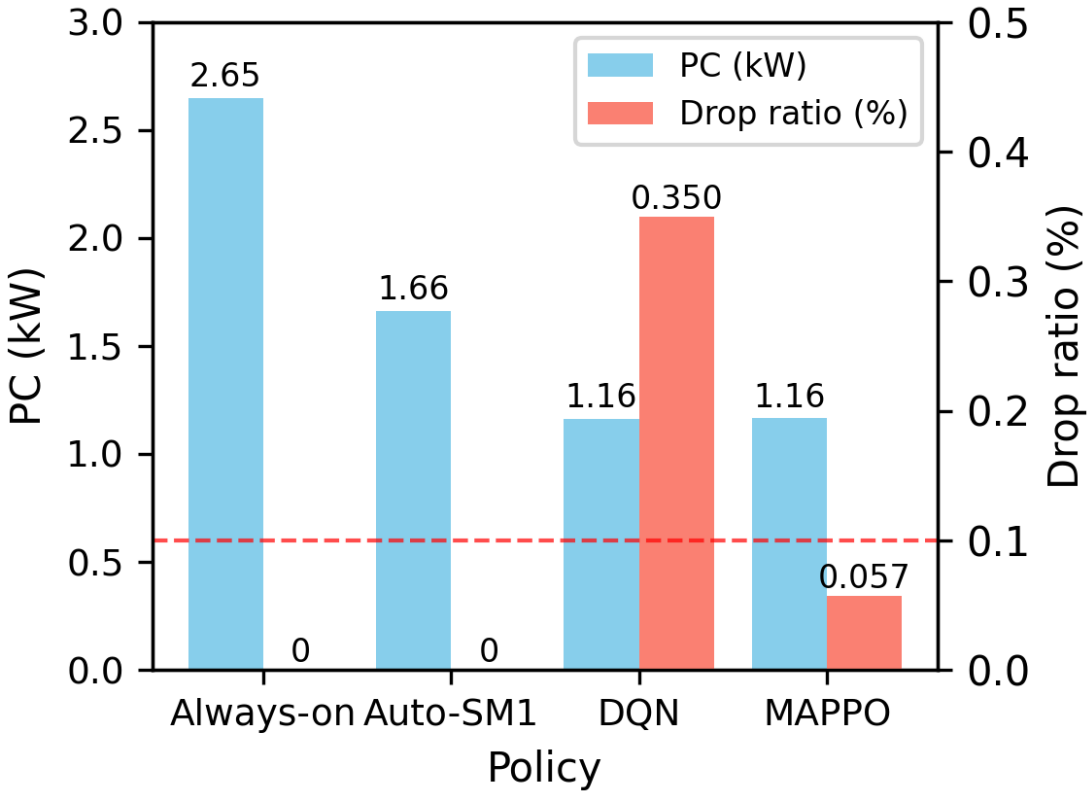






Simulation Results

- ***MAPPO*** demonstrates superior energy efficiency compared to the non-learning baselines, achieving a 56.23% reduction in relative to the ***Always-on*** and a 30.12% reduction compared to ***Auto-SM1***.



Thank you