Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 12, No. 01, March 2026.

A SCALABLE FEDERATED DEEP REINFORCEMENT LEARNING ARCHITECTURE FOR COLLABORATIVE LEARNING

Tarek Amine Haddad

(Received: 29-Dec.-2025, Revised: 8-Feb.-2026, Accepted: 10-Feb.-2026)

ABSTRACT

Federated Learning enables collaborative model training without sharing raw data, while Deep Reinforcement Learning provides powerful mechanisms for sequential decision-making. However, their integration suffers from limited scalability, sensitivity to non-IID data and unstable convergence in distributed environments. This paper proposes a Scalable Federated Deep Reinforcement Learning (SFDRL) architecture in which distributed agents learn local policies and periodically contribute to a global model via an adaptive, performance-aware aggregation strategy. Unlike conventional FedRL methods that rely on uniform averaging, SFDRL weights local updates according to their learning effectiveness, resulting in faster convergence and improved stability under heterogeneous data distributions. In addition, a selective communication mechanism is introduced to reduce communication overhead by up to 28% *and* 64% *compared with FedAvg and FedRL, respectively. Extensive experiments demonstrate that SFDRL outperforms compared methods, achieving higher cumulative rewards, reduced variance during training and improved scalability in large-scale distributed settings. These results confirm the suitability of SFDRL for practical deployment in distributed intelligent systems.*

KEYWORDS

Federated learning, Deep reinforcement learning, Collaborative learning, Distributed intelligence, Scalability, Adaptive aggregation.

1. INTRODUCTION

Deep Reinforcement Learning (DRL) has emerged as a powerful paradigm for sequential decision-making in complex and high-dimensional environments. By integrating reinforcement learning principles with deep neural networks, DRL enables agents to learn optimal control policies directly from raw sensory inputs without relying on handcrafted features. This capability has led to remarkable successes in a wide range of applications, including robotics, autonomous driving, intelligent transportation systems and resource management. Value-based methods, such as Deep Q-Networks (DQNs) and policy-based or actor-critic approaches like DDPG and PPO, have demonstrated strong performance in both discrete and continuous action spaces. Despite these advances, conventional DRL typically relies on centralized training with full access to experience data, which limits its scalability and raises privacy and communication concerns in distributed and multi-agent environments [1]. In addition, DRL has demonstrated remarkable success in various domains, including robotics [2], intelligent transportation [3] and autonomous systems [4]. By combining deep neural networks with reinforcement learning, DRL enables agents to learn complex policies directly from high-dimensional state spaces [17]. However, conventional DRL approaches typically require centralized data collection, which can be impractical or undesirable in distributed and privacy-sensitive environments.

Federated learning (FL) has emerged as a promising solution to train machine-learning models collaboratively without sharing raw data [5], [8]. In FL, multiple agents or clients train local models on their own data and periodically aggregate updates into a global model, preserving data privacy while leveraging collective knowledge. Integrating FL with DRL enables multiple agents to learn collaboratively in a distributed setting, but it introduces challenges, such as non-IID data, communication constraints and unstable policy aggregation [6]-[7].

Recent studies have attempted to address these challenges by applying standard federated averaging (FedAvg) to DRL agents [9]-[10], but performance often degrades in heterogeneous environments due to divergent local updates. Additionally, excessive communication overhead can limit scalability in

T.A. Haddad is with LEREESI Laboratory, Higher National School of Renewable Energies, Environment and Sustainable Development, Batna 05078, Algeria. Email: tarek.haddad@hns-re2sd.dz

large-scale multi-agent systems [11]-[12]. These limitations motivate the development of a scalable, stable and communication-efficient federated DRL framework.

Recent advancements in federated reinforcement learning (FedRL) have focused on improving communication efficiency, scalability and learning stability in distributed environments. Di et al. [13] proposed a FedRL-based recommender system that leverages a reinforcement selector and hypernet generator to reduce communication overhead. Zhang et al. [9] introduced a multi-agent approach to optimize federated learning in industrial IoT systems, highlighting the challenges of heterogeneous clients. Pan et al. [11] developed RFCSC, which combines dynamic client selection with adaptive gradient compression for communication-efficient reinforcement learning. Pinto Neto et al. [12] provided a comprehensive survey on FedRL applications in IoT, discussing opportunities and open challenges in privacy-preserving distributed learning. These studies motivate the development of scalable and stable frameworks, like SFDRL, that address both communication and heterogeneity challenges in multi-agent reinforcement learning.

In this paper, we propose Scalable Federated Deep Reinforcement Learning (SFDRL), a collaborative learning framework in which distributed agents perform local DRL training and periodically synchronize with a global model through federated coordination. SFDRL incorporates an adaptive aggregation strategy that weights local updates according to learning performance and stability, as well as a selective participation mechanism that allows only informative agents to communicate, thereby improving scalability and reducing communication overhead in heterogeneous environments. The key contributions of this work are:

- We design an adaptive aggregation mechanism that weights local model updates based on performance and stability, mitigating the effects of non-IID data and unstable learning.
- We introduce selective participation, allowing only agents with significant local improvements to communicate updates, reducing communication overhead while maintaining learning efficiency.
- We provide a comprehensive experimental evaluation in heterogeneous multi-agent environments, demonstrating that SFDRL achieves near-centralized DRL performance with significantly lower communication cost.
- We conduct ablation studies to validate the effectiveness of adaptive aggregation and selective participation in improving stability and scalability.

The remainder of this paper is organized as follows. Section 2 reviews related work on federated reinforcement learning. Section 3 formulates the problem and Section 4 presents the proposed SFDRL algorithm. Sections 5 and 6 provide theoretical analysis and experimental setup, respectively. Section 7 presents results and discussion, including ablation studies. Finally, Section 8 concludes the paper and outlines future research directions.

2. RELATED WORK

Deep Reinforcement Learning (DRL) has achieved significant success in domains, such as robotics, autonomous systems and intelligent transportation [2][3][4]. By combining deep neural networks with reinforcement learning, DRL agents can learn complex policies directly from high-dimensional state spaces. However, conventional DRL typically relies on centralized training and full access to all experience data, which limits its applicability in distributed or privacy-sensitive environments [18].

Federated Learning (FL) enables collaborative training across multiple clients without sharing raw data [5, 8]. In FL, clients train local models and periodically aggregate updates to a global model, preserving privacy while leveraging distributed knowledge. Standard FL approaches, such as FedAvg, face challenges with non-IID data, heterogeneous clients and limited communication bandwidth [7][9]. Truex et al. [16] proposed a hybrid privacy-preserving federated learning approach that combines differential privacy with secure multi-party computation to protect against inference attacks on both exchanged messages and the final model, achieving scalable and accurate training.

Federated Reinforcement Learning (FedRL) integrates DRL and FL to allow multiple agents to learn collaboratively in distributed environments. Early FedRL approaches applied FedAvg to DRL agents, but performance often degrades in heterogeneous settings due to divergent local updates [9][13].

Communication overhead is also a major limitation in large-scale multi-agent systems. Recent advances in federated learning have also explored its application to recommender systems and personalized learning tasks. For example, federated recommender systems have been proposed that leverage diffusion augmentation and guided denoising to enhance recommendation quality under privacy constraints [21].

Recent studies have proposed various strategies to address these challenges. Di et al. [13] introduced a FedRL-based recommender system using a reinforcement selector and hypernet generator to reduce communication. Tian et al. [22] proposed FDDL, a framework that leverages deep reinforcement learning for cache admission and federated learning for parameter sharing, resulting in higher cache hit ratios and lower communication costs compared to conventional and other DRL-based caching schemes. CU-BIC-Learn introduces a reinforcement learning-based enhancement to the CUBIC congestion-control algorithm by using Q-learning to adapt congestion window thresholds based on network feedback [23]. Simulation results show significant improvements in packet loss, bandwidth utilization, latency and fairness compared to standard CUBIC and other classical congestion control schemes.

Communication overhead remains a critical bottleneck in FedRL, particularly for large-scale multi-agent systems. Strategies, such as selective participation, adaptive aggregation and gradient compression have been explored to reduce communication while maintaining learning performance [11], [14]. Zhang et al. [9] demonstrated that multi-agent approaches with optimized client selection can improve both convergence and efficiency in industrial IoT applications. These insights motivate the design of SFDRL, which combines adaptive aggregation and selective participation to achieve near-centralized performance while ensuring scalability and privacy. In addition to communication-efficient strategies, incentive mechanisms for resource-limited devices in federated learning have also been explored. Zhao et al. [15] proposed a learning-based multi-task federated edge learning (FEL) mechanism that jointly designs economic incentives and participation contribution strategies.

In summary, prior work has explored federated learning, reinforcement learning and their integration in distributed and heterogeneous environments. However, existing approaches often either suffer from high communication overhead or reduced learning stability. SFDRL is designed to bridge this gap, providing a scalable, stable and communication-efficient framework for federated multi-agent reinforcement learning.

3. PROBLEM FORMULATION

We consider a collaborative-learning system composed of a set of distributed agents $\mathcal{N} = \{1,2,\dots,N\}$, where each agent interacts with its own local environment and aims to learn an optimal decision-making policy through reinforcement learning. Due to privacy, communication or ownership constraints, raw interaction data cannot be shared among agents. Instead, learning is performed in a federated manner by exchanging model parameters with a coordinating server.

Each agent $i \in \mathcal{N}$ is modeled as a Markov Decision Process (MDP) defined by the tuple $\langle \mathcal{S}_i, \mathcal{A}_i, \mathcal{P}_i, r_i, \gamma \rangle$, where $\mathcal{S}_i$ and $\mathcal{A}_i$ denote the state and action spaces, respectively, $\mathcal{P}_i(s' \mid s,a)$ represents the state-transition probability, $r_i(s,a)$ is the local reward function and $\gamma \in (0,1)$ is the discount factor. The environments may differ across agents, leading to heterogeneous and non-identically distributed (non-IID) data.

Each agent seeks to learn a parameterized policy $\pi_{\theta_i}(a \mid s)$ (or an action-value function $Q_{\theta_i}(s,a)$) that maximizes its expected cumulative discounted return:

$$J_i(\theta_i) = \mathbb{E}_{\pi_{\theta_i}}\left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t,a_t)\right]. \quad (1)$$

In a centralized reinforcement-learning setting, all experiences would be aggregated to optimize a single global model. However, such an approach is impractical in distributed systems due to privacy constraints and communication overhead. To address this, we adopt a federated-learning paradigm in which each agent performs local training and periodically communicates model parameters instead of raw data.

Let θ^t denote the global model parameters at federated-communication round t. At the beginning of each round, the server broadcasts θ^t to a sub-set of participating agents. Each selected agent initializes its local model as $\theta_i^t \leftarrow \theta^t$ and performs E local reinforcement-learning updates through interaction

with its environment, producing updated parameters θ_i^{t+1}. The local optimization process can be expressed as:

$$\theta_i^{t+1} = \theta^t - \eta \nabla_\theta \mathcal{L}_i(\theta), \tag{2}$$

where η is the learning rate and $\mathcal{L}_i(\theta)$ denotes the local DRL loss function derived from temporal-difference or policy-gradient updates. Eq. (2) defines the local update of agent i during federated-communication round t. Here, θ_i^{t+1} represents the agent's updated local model parameters after performing E reinforcement-learning steps. The term η denotes the learning rate controlling the step size of each update, while $\nabla_\theta \mathcal{L}_i(\theta)$ is the gradient of the local DRL loss function $\mathcal{L}_i(\theta)$, which can be computed using temporal-difference (TD) or policy-gradient methods depending on the chosen DRL algorithm. This formulation ensures that each agent adjusts its local policy toward minimizing its own expected loss while preserving privacy, as only model parameters - not raw experience data - are communicated to the server. Subsequently, these local updates are aggregated at the central server to refine the global policy θ^{t+1}, which is then broadcast to agents in the next communication round.

After local training, participating agents transmit their updated parameters to the server. The objective of the federated-aggregation process is to compute a global model that reflects the collective learning progress while accounting for heterogeneity and training stability. Formally, the global aggregation can be written as:

$$\theta^{t+1} = \sum_{i \in \mathcal{N}_t} w_i^t \theta_i^{t+1} \tag{3}$$

where $\mathcal{N}_t \subseteq \mathcal{N}$ denotes the set of participating agents at round t and w_i^t represents the aggregation weight associated with agent i, satisfying $\sum_{i \in \mathcal{N}_t} w_i^t = 1$.

Unlike conventional federated averaging, the weights w_i^t are not solely determined by data volume but are designed to reflect the contribution quality of each agent. In particular, they may depend on performance indicators, such as the average episodic return, training stability or improvement magnitude observed during local learning. This formulation allows the global model to emphasize informative and reliable updates while reducing the influence of noisy or unstable ones.

The overall objective of the proposed federated deep reinforcement-learning framework is to learn a global policy parameter vector θ^* that maximizes the aggregated expected return across all agents:

$$\theta^* = \arg\max_\theta \sum_{i \in \mathcal{N}} \mathbb{E}_{\pi_\theta} \left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t, a_t) \right], \tag{4}$$

Subject to decentralized data constraints and limited communication, this formulation captures the fundamental trade-off between collaborative performance, scalability and privacy preservation and serves as the basis for the proposed scalable federated deep reinforcement-learning architecture.

4. PROPOSED METHOD

This section presents the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) method, which enables efficient and robust collaborative learning among distributed agents operating in heterogeneous environments. The proposed approach integrates federated learning with deep reinforcement learning through adaptive-aggregation and selective-participation mechanisms, aiming to improve scalability, stability and learning efficiency under non-IID conditions.

At the beginning of each federated-communication round t, a global model parameterized by θ^t is maintained by a coordinating server. A sub-set of agents $\mathcal{N}_t \subseteq \mathcal{N}$ is selected to participate in the current round. The server broadcasts θ^t to the selected agents, which initialize their local models accordingly. Each agent then interacts with its local environment and performs multiple reinforcement-learning updates using its private experience. The proposed framework is model-agnostic and can be instantiated with either value-based or actor-critic DRL algorithms.

During local training, agent i updates its parameters by minimizing a reinforcement-learning loss function derived from temporal-difference or policy-gradient learning. After E local training episodes,

the agent obtains an updated parameter vector θ_i^{t+1} together with performance indicators reflecting the quality of its learning process. These indicators include the average episodic return $\bar{R}_i^t$ and a stability measure σ_i^t, computed as the variance of recent returns. Unlike conventional federated learning, where all clients contribute equally or proportionally to data size, the proposed method evaluates the reliability and usefulness of each update before aggregation.

To address heterogeneity and unstable learning dynamics, an adaptive weighting mechanism is introduced. Each participating agent is assigned a contribution weight w_i^t defined as

$$w_i^t = \frac{\phi\left(\bar{R}_i^t, \sigma_i^t\right)}{\sum_{j \in \mathcal{N}_t} \phi\left(\bar{R}_j^t, \sigma_j^t\right)}, \tag{5}$$

where $\phi(\cdot)$ is a monotonically increasing function with respect to performance and a decreasing function with respect to instability. This design favors agents that exhibit consistent learning progress while reducing the influence of noisy or poorly converged updates. As a result, the aggregation process becomes more robust to non-IIID data distributions and heterogeneous environments.

The global model is updated through a weighted aggregation of local parameters:

$$\theta^{t+1} = \sum_{i \in \mathcal{N}_t} w_i^t \theta_i^{t+1}. \tag{6}$$

This adaptive aggregation enables the global policy to capture shared knowledge across agents while mitigating divergence caused by conflicting local objectives.

To further enhance scalability, the proposed framework incorporates a selective participation mechanism that limits unnecessary communication. Each agent evaluates the significance of its update by measuring the relative improvement in performance between consecutive rounds. Only agents the improvement of which exceeds a predefined threshold ϵ are allowed to transmit their model updates. Formally, agent i participates in round t if:

$$\left|\bar{R}_i^t - \bar{R}_i^{t-1}\right| \geq \epsilon. \tag{7}$$

This mechanism reduces communication overhead and alleviates network congestion, while preserving learning effectiveness by prioritizing informative updates.

The overall training procedure alternates between local reinforcement learning and federated coordination until convergence or a maximum number of communication rounds is reached. Through adaptive aggregation and selective participation, the proposed SFDRL framework achieves improved stability, faster convergence and enhanced scalability compared with standard federated reinforcement-learning approaches. The method preserves data locality and supports heterogeneous agents, making it suitable for large-scale collaborative-learning systems.

4.1 Overall Architecture

The overall architecture of the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) framework is illustrated in Figure 1. The architecture is designed to enable efficient, privacy-preserving and scalable collaborative learning across distributed agents by integrating federated learning, multi-agent reinforcement learning and incentive-aware coordination mechanisms.

The system is composed of three main layers: edge devices, edge servers and a central aggregator. At the edge level, heterogeneous devices (e.g., IoT nodes, smartphones or sensors) perform local interactions with their environments and collect state information. Local training is conducted without sharing raw data, ensuring data privacy. Each device computes local policy updates or state representations and transmits only the necessary model-related information to the upper layer.

At the edge-server level, a multi-agent learning module coordinates the received local information. This layer is responsible for managing multiple agents, handling heterogeneous data distributions and executing collaborative learning through shared representations. An incentive mechanism is integrated to encourage active participation of resource-constrained devices by assigning adaptive rewards based on their contributions. The reward feedback plays a key role in stabilizing training and improving long-term participation. The edge servers also estimate training ratios and intermediate policies that guide

local learning behavior.

The central aggregator performs global model aggregation and policy optimization. It collects model updates or policy parameters from edge servers and aggregates them using a federated strategy to produce a global policy. This global policy is then redistributed to the edge servers, closing the learning loop. The aggregation process ensures scalability while reducing communication overhead and preserving privacy.

At the core of the framework lies the proposed SFDRL algorithm, which coordinates reward feedback, aggregation and policy updates across all layers. By jointly optimizing local learning, incentive allocation and global aggregation, SFDRL enables efficient multi-agent collaboration under heterogeneous and communication-constrained environments.

The flowchart in Figure 2 illustrates the main steps of the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) algorithm. The process begins with the initialization of the global policy and agent networks. The global policy is then distributed to edge servers and agents for local training. At the edge-device level, each agent interacts with its environment by observing states, taking actions, receiving rewards and updating its local policy.

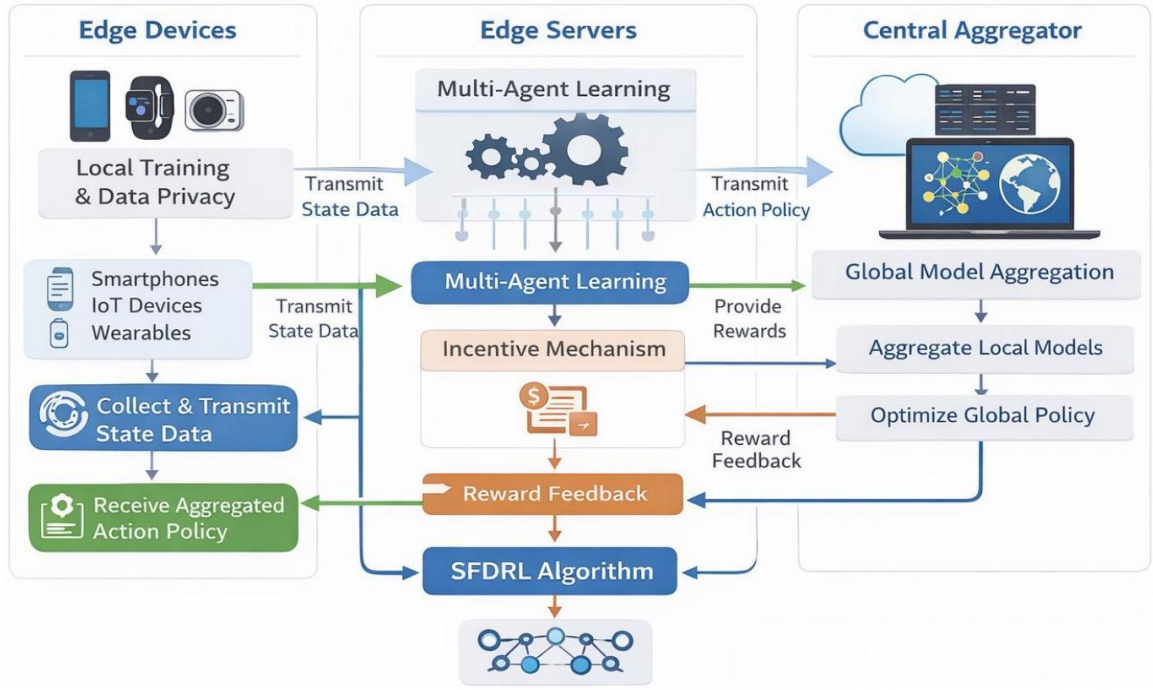

Figure 1. Overall architecture of the proposed SFDRL framework. The system integrates edge devices, edge servers and a central aggregator to enable scalable and privacy-preserving federated deep reinforcement learning with incentive-aware coordination.

Following local training, edge server aggregation collects the local updates from participating devices, optionally applies selective participation and aggregates the results into an edge-level model. The central aggregator then collects edge-level models to update the global policy, which is redistributed to the edge servers, completing the collaborative learning loop. A decision point checks whether the maximum number of episodes is reached or convergence is achieved; if not, the process repeats. This design ensures scalable, communication-efficient and privacy-preserving federated reinforcement learning across heterogeneous multi-agent environments.

4.2 Algorithm Description

The proposed Scalable Federated Deep Reinforcement Learning (SFDRL) algorithm integrates local reinforcement learning with federated coordination to enable collaborative policy optimization across distributed agents while preserving data privacy. At each federated communication round, a sub-set of agents is selected to participate and the current global model parameters are broadcast to them. Each agent initializes its local model with the received parameters and interacts with its environment to collect

experience. The local model is updated using standard DRL optimization techniques, such as temporal-difference learning for value-based methods or policy-gradient updates for actor-critic algorithms.

After completing local training episodes, each agent computes performance indicators, including the average episodic return and a stability measure. These indicators are used to determine whether the agent's update is informative enough to contribute to the global model, as governed by the selective participation threshold. Only agents the local updates of which exceed the threshold transmit their parameters to the server, which significantly reduces communication overhead.

The server aggregates the received local updates using an adaptive weighting strategy, in which each agent's contribution is proportional to both the quality and stability of its learning progress. This aggregation produces an updated global model that reflects the collective knowledge of the agents while mitigating the influence of noisy or unstable updates. The global model is then redistributed to the agents in the next round and the process repeats until convergence or until a maximum number of communication rounds is reached.

Overall, the SFDRL algorithm balances exploration and exploitation in local environments, ensures scalability through selective participation and enhances robustness *via* adaptive aggregation. To facilitate adoption, we provide step-by-step instructions for implementing SFDRL. Algorithm 1 summarizes the training loop and provides a step-by-step summary of the complete procedure, while Table 1 provides suggested default hyper-parameters for stable performance across heterogeneous environments.

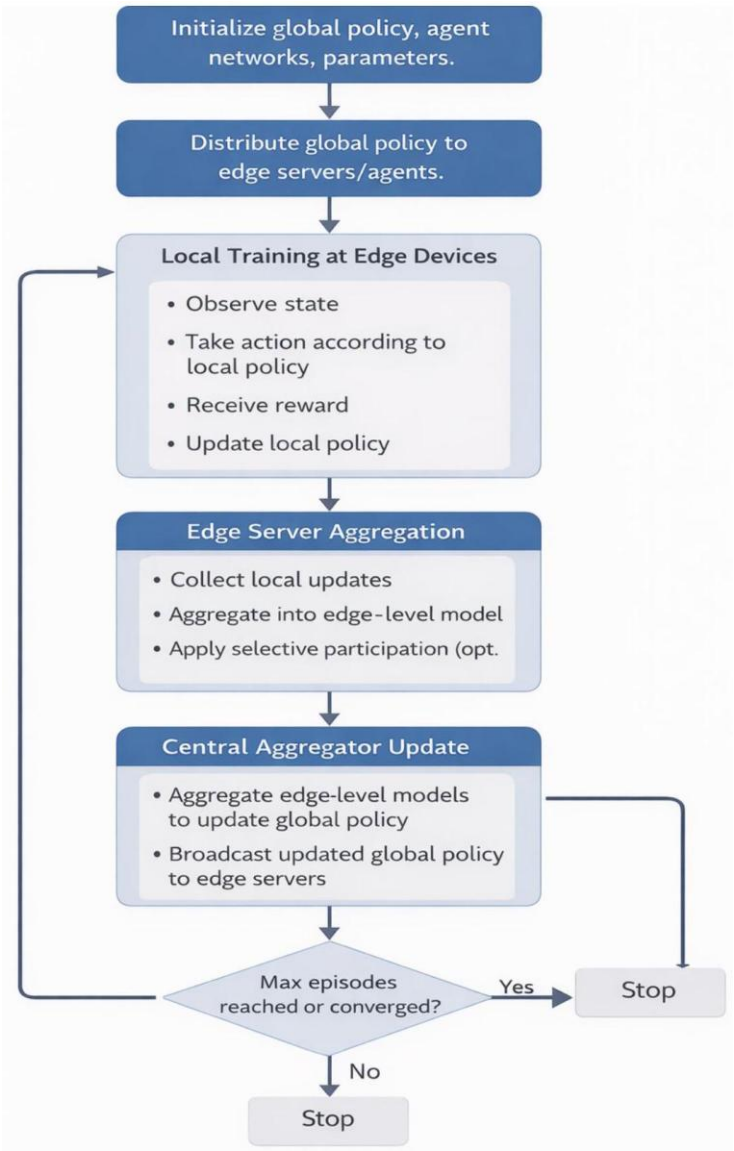

Figure 2. Flowchart of the SFDRL algorithm.

5. THEORETICAL ANALYSIS

In this section, we analyze the theoretical properties of the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) framework, focusing on convergence, stability and communication efficiency in distributed environments. While a formal proof of convergence for general DRL is difficult due to the non-convexity of neural networks and stochastic environment dynamics, we provide intuitive arguments and bounds based on existing federated and reinforcement-learning theory.

Algorithm 1 Scalable Federated Deep Reinforcement Learning (SFDRL)

Input: Number of agents N; communication rounds T; local training episodes E; learning rate η; discount factor γ; participation threshold ϵ

Output: Global policy parameters θ^T

Initialize global model parameters θ^0 randomly

for $t = 0$ **to** $T-1$ **do**

  Server selects a subset of agents $\mathcal{N}_t \subseteq \mathcal{N}$ Server broadcasts global parameters θ^t to all $i \in \mathcal{N}_t$

  foreach *agent* $i \in \mathcal{N}_t$ ***in parallel*** **do**

    Initialize local parameters $\theta_i \leftarrow \theta^t$

    for $e = 1$ **to** E **do**

      Interact with local environment using policy π_{θ_i} Collect transitions (s, a, r, s') Update θ_i using DRL optimization step

    Compute average episodic return $\bar{R}_i^t$ Compute stability metric σ_i^t

    if $|\bar{R}_i^t - \bar{R}_i^{t-1}| \geq \epsilon$ **then**

      Send $(\theta_i, \bar{R}_i^t, \sigma_i^t)$ to server

  Server computes adaptive aggregation weights:

$$w_i^t = \frac{\phi(\bar{R}_i^t, \sigma_i^t)}{\sum_{j \in \mathcal{N}_t} \phi(\bar{R}_j^t, \sigma_j^t)}$$

  Update global model:

$$\theta^{t+1} = \sum_{i \in \mathcal{N}_t} w_i^t \theta_i$$

return θ^T

5.1 Convergence Intuition

Each agent i performs local reinforcement-learning updates that aim to maximize its expected cumulative return:

$$J_i(\theta_i) = \mathbb{E}_{\pi_{\theta_i}}\left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t, a_t)\right] \tag{8}$$

Under standard DRL assumptions (bounded rewards, learning rate η sufficiently small and sufficient exploration), the local update steps converge to a local optimum of $J_i(\theta_i)$.

The adaptive weighted aggregation at the server ensures that the global model θ^t is a convex combination of stable local updates:

$$\theta^{t+1} = \sum_{i \in \mathcal{N}_t} w_i^t \theta_i^{t+1}, \sum_i w_i^t = 1 \tag{9}$$

By prioritizing updates with high performance and low instability, the aggregation mitigates divergence caused by non-IID environments. Therefore, the global policy progressively approaches a consensus that reflects the collective intelligence of all agents, improving convergence in heterogeneous settings compared with naive federated averaging. Nevertheless, the presented analysis provides a reasonable approximation of the learning dynamics and offers theoretical intuition that is consistent with the empirical results observed in heterogeneous and dynamic experimental settings.

5.2 Stability Analysis

Stability in SFDRL is influenced by two factors: variance in local learning and heterogeneity among agent environments. The weighting function $\phi(\bar{R}_i^t, \sigma_i^t)$ reduces the impact of unstable updates (high σ_i^t) while amplifying reliable contributions. Let $\Delta\theta_i^t$ denote the update magnitude for agent i. The expected deviation of the global model after aggregation can be bounded as:

Table 1. Hyper-parameters for SFDRL implementation.

Parameter	Value	Description
Learning Rate (η)	0.001-0.01	Step size for local DRL updates
Local Updates (E)	10	Number of local training iterations per round
Batch Size	32-128	Number of experiences per gradient update
Discount Factor (γ)	0.99	Weighting of future rewards
Participation Rate (p_t)	0.5-1.0	Fraction of agents participating per round
Aggregation Weighting	Adaptive	Weight local updates based on performance
Communication Rounds (T)	100-500	Total number of federated rounds
Exploration Rate (ϵ)	0.01	ϵ-greedy exploration parameter for DRL

$$\|\theta^{t+1} - \theta^t\| \leq \sum_{i \in \mathcal{N}_t} w_i^t \left\|\Delta\theta_i^t\right\| \tag{10}$$

Since unstable updates receive lower weights, large fluctuations are suppressed, leading to a smoother global learning trajectory and improved stability over time.

5.3 Communication Complexity

In standard federated reinforcement learning, all agents communicate updates at every round, leading to high communication costs proportional to $N \cdot T \cdot |\theta|$, where $|\theta|$ is the model size. SFDRL reduces communication *via* selective participation: only agents with meaningful local improvements above a threshold ϵ transmit updates. Denoting p_t as the fraction of participating agents at round t, the total communication complexity becomes:

$$\mathcal{O}\left(|\theta| \sum_{t=1}^{T} p_t N\right), p_t \leq 1, \tag{11}$$

which can be substantially smaller than the naive approach, particularly in large-scale systems with sparse significant updates.

5.4 Discussion

The combination of adaptive aggregation and selective participation provides a theoretical rationale for the observed empirical improvements in convergence and stability. By emphasizing informative updates and suppressing noisy contributions, SFDRL mitigates common challenges in federated reinforcement learning, including non-IID data distributions, heterogeneous agent behaviors and unstable local learning dynamics. Moreover, selective communication ensures scalability without compromising the global learning quality. It is worth noting that the theoretical analysis presented in this section relies on standard assumptions commonly adopted in deep reinforcement learning and federated learning, such as bounded rewards, sufficient exploration and relatively stable local update dynamics. While these assumptions facilitate tractable analysis and provide useful insights into convergence behavior, they may not fully capture the complexity of highly dynamic or non-stationary environments. Extending the theoretical framework to relax these assumptions, for example by explicitly modeling environment dynamics, asynchronous updates or time-varying participation, constitutes an important direction for future work and could further strengthen the robustness of the proposed SFDRL framework.

Overall, the theoretical analysis demonstrates that SFDRL is well-suited for large-scale, distributed and privacy-preserving reinforcement-learning systems.

6. EXPERIMENTAL SETUP

To evaluate the effectiveness of the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) framework, we design experiments that assess convergence, stability, scalability and communication efficiency in distributed environments. The evaluation includes comparisons with

centralized DRL and standard federated reinforcement-learning baselines. The chosen simulation-based evaluation allows systematic analysis of scalability and communication efficiency, which would be difficult to isolate in uncontrolled real-world settings.

6.1 Environment and Agents

Experiments are conducted in a set of simulated environments that represent heterogeneous, non-IID scenarios. Each agent $i \in \mathcal{N}$ interacts with a local environment characterized by state space $\mathcal{S}_i$ and action space $\mathcal{A}_i$, as defined in Section 3. Agents receive local rewards $r_i(s, a)$ and update their policies using standard DRL algorithms, including DDPG for continuous action spaces and DQN for discrete action spaces. To assess generality, multiple environments are configured with varying dynamics, stochastic transitions and reward functions.

6.2 Baseline Methods

We compare SFDRL against the following approaches:

- **Centralized DRL** [19]: All agent experiences are aggregated centrally and a single global model is trained. This serves as an upper-bound performance reference, but assumes full data sharing.
- **Independent DRL (IDRL)** [20]: Each agent trains its local DRL model without any collaboration. This highlights the benefits of federated coordination.
- **Federated DRL with Standard FedAvg** [7]: Local DRL models are trained independently and aggregated using conventional federated averaging without adaptive weighting or selective participation.
- **Federated Reinforcement Learning (FedRL)** [13]: Federated Reinforcement Learning (FedRL) is defined as a collaborative learning framework in which multiple agents independently interact with their environments and train local reinforcement-learning models, while a central server periodically aggregates model updates to form a global policy without requiring the sharing of raw data.

In addition to the selected baselines, several recent state-of-the-art federated learning and deep reinforcement-learning methods could be considered for comparison, including communication-efficient federated-optimization techniques, trust-aware or uncertainty-aware aggregation strategies and asynchronous federated reinforcement learning frameworks. However, many of these approaches are designed for supervised or static learning settings or require problem-specific assumptions that make direct and fair comparison with SFDRL challenging. The baselines adopted in this study represent widely used and representative methods in federated and reinforcement learning, providing a meaningful and fair evaluation of the proposed framework.

6.3 Evaluation Metrics

To comprehensively assess the performance of SFDRL and baseline methods, we employ the following metrics:

1) **Cumulative Reward** (R_{cum}): The average episodic return achieved by the global policy across all agents and episodes. Formally,

$$R_{cum} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T} r_i^t, \tag{12}$$

where N is the number of agents, T is the number of time steps per episode and r_i^t is the reward received by agent i at time t.

2) **Convergence Speed** (C_s): The number of communication rounds required for the global policy to reach a predefined reward threshold R_{th}:

$$C_s = \min\{t: R_{cum}^t \geq R_{th}\}, \tag{13}$$

3) **Stability** (σ^2): Variance of episodic returns over communication rounds:

$$\sigma^2 = \frac{1}{T}\sum_{t=1}^{T} (R_{cum}^t - \bar{R}_{cum})^2, \tag{14}$$

where $\bar{R}_{\text{cum}}$ is the mean cumulative reward over T rounds.

4) **Communication Cost** (C_{comm}) : Total number of model updates transmitted from agents to the server:

$$C_{\text{comm}} = \sum_{t=1}^{T} \sum_{i \in \mathcal{S}_t} \text{size}\left(\theta_i^t\right) \tag{15}$$

where $\mathcal{S}_t$ is the set of agents participating in round t and $\text{size}\left(\theta_i^t\right)$ is the size of the transmitted model.

6.4 Implementation Details

The experiments are implemented in Python using PyTorch. Agents train in parallel on multiple CPU cores and communicate with a central server simulated in the same process. Neural networks for value and policy functions consist of two hidden layers with 128 neurons each, ReLU activation and Adam optimizer with a learning rate $\eta = 0.001$. Discount factor is set to $\gamma = 0.99$ and each communication round consists of $E = 10$ local training episodes. The selective participation threshold ϵ is empirically set to 0.01 to balance communication efficiency and learning performance. Each experiment is repeated 10 times with different random seeds to account for stochasticity.

6.5 Experimental Procedure

At the beginning of each round, the server broadcasts the global model to selected agents. Agents perform local DRL training, compute performance metrics and decide whether to participate in aggregation based on the selective-participation criterion. The server aggregates updates using the adaptive-weighting scheme described in Section 4. The process continues for T communication rounds or until convergence is achieved. All baseline methods follow the same evaluation procedure for a fair comparison.

7. RESULTS AND DISCUSSION

In this section, we present the experimental results of the proposed Scalable Federated Deep Reinforcement Learning (SFDRL) framework and compare its performance with those of baseline methods. We analyze learning efficiency, convergence behavior, stability, scalability and communication efficiency to demonstrate the advantages of our approach.

7.1 Learning Performance

Figure 3 shows the cumulative reward over communication rounds for SFDRL, centralized DRL, independent DRL (IDRL) and standard federated DRL with FedAvg. SFDRL achieves faster convergence and higher final rewards than IDRL and FedAvg, approaching the performance of centralized DRL without sharing raw data. The adaptive aggregation mechanism allows SFDRL to leverage high-quality local updates, leading to improved learning efficiency across heterogeneous agents.

7.2 Convergence and Stability Analysis

Figure 4 summarizes the variance of episodic returns for all methods. SFDRL exhibits lower variance compared with FedAvg and IDRL, demonstrating enhanced stability during training. The adaptive weighting reduces the influence of unstable or poorly performing agents, resulting in smoother global learning trajectories. It is observed that centralized DRL often achieves higher cumulative rewards compared to federated methods, including SFDRL. This performance advantage arises primarily because centralized training has access to the full set of experiences from all agents, enabling the model to learn from the complete state-action distribution. In contrast, federated approaches operate on local data, which may be heterogeneous and non-IID, leading to incomplete or biased learning. Furthermore, centralized DRL updates the model continuously without communication constraints, avoiding delays and aggregation approximations inherent in federated learning. Aggregating local policies in FedAvg or even in SFDRL can introduce inconsistencies when local updates diverge, slightly reducing global-policy performance. Despite this gap, SFDRL narrows the difference by leveraging adaptive aggregation and selective participation, achieving near-centralized performance while maintaining data privacy, reducing communication overhead and enabling scalable multi-agent deployment.

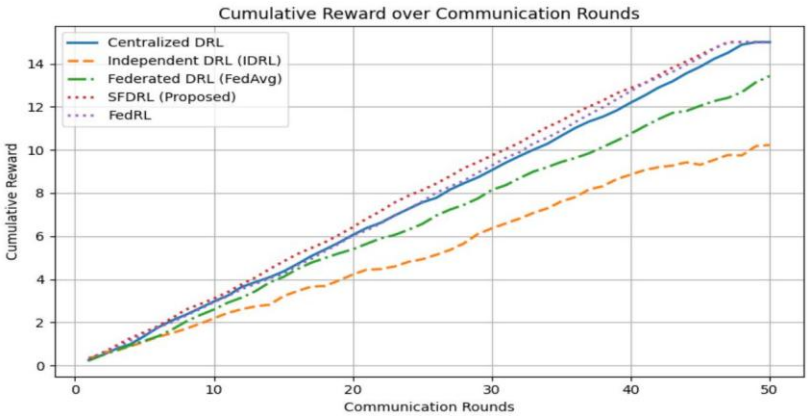

Figure 3. Cumulative reward over federated communication rounds for different methods. SFDRL converges faster and achieves higher returns than baseline federated and independent DRL.

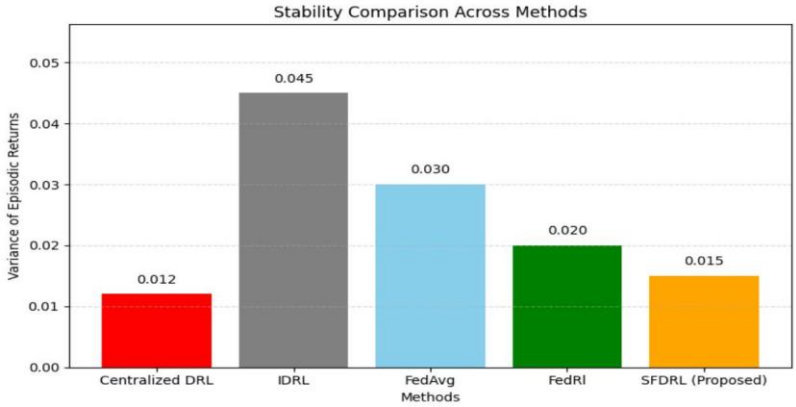

Figure 4. Variance of episodic returns for different methods. Lower variance indicates higher stability.

7.3 Communication Efficiency

Figure 5 presents the total number of transmitted updates during training. SFDRL significantly reduces communication overhead due to selective participation. Only agents with meaningful improvements transmit updates, decreasing redundant transmissions while maintaining performance comparable to full participation federated learning.

7.4 Scalability Analysis

To evaluate scalability, experiments were conducted with varying numbers of agents $N = \{10,20,50,100\}$. SFDRL maintains stable convergence and competitive cumulative rewards as the number of agents increases, whereas FedAvg performance degrades slightly in highly heterogeneous settings. The selective-participation mechanism ensures that only informative updates are aggregated, preventing communication bottlenecks and maintaining learning quality in large-scale deployments.

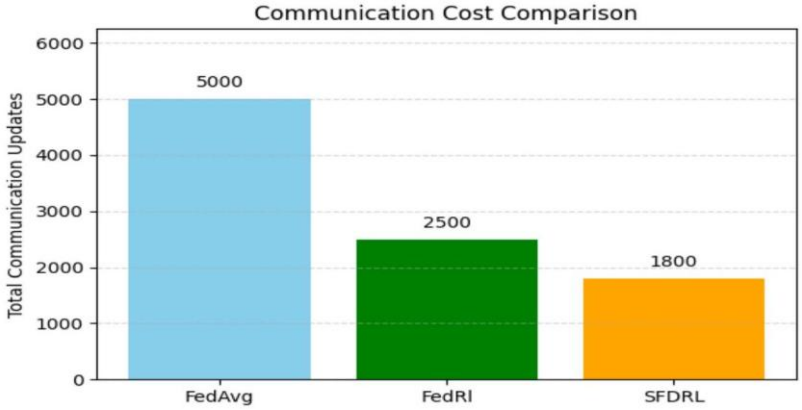

Figure 5. Total communication cost for different federated-learning methods. SFDRL reduces communication overhead while preserving learning efficiency.

7.5 Discussion

The experimental results validate the effectiveness of SFDRL in distributed, heterogeneous environments. Key observations include:

- Adaptive aggregation improves convergence speed and stability by weighting agent contributions according to performance and reliability.
- Selective participation significantly reduces communication costs without sacrificing learning performance.
- SFDRL scales well with the number of agents and is robust to non-IID data distributions.
- Compared with centralized DRL, SFDRL achieves near-optimal performance while preserving data privacy and agent autonomy.

Overall, the results demonstrate that SFDRL provides a practical and efficient framework for large-scale collaborative reinforcement-learning systems. While SFDRL is compared against standard and widely adopted federated and reinforcement-learning baselines, future work will include extensive comparisons with emerging state-of-the-art methods, such as asynchronous and communication-efficient federated DRL frameworks. This will further validate the generality and robustness of SFDRL across diverse learning paradigms.

Although SFDRL preserves data locality by design, potential privacy and security risks remain, as in most federated-learning frameworks. Model updates exchanged during training may leak sensitive information through inference or poisoning attacks. To mitigate these risks, SFDRL can be naturally combined with established privacy-preserving and security mechanisms, such as secure aggregation, differential privacy and robust aggregation strategies. Secure aggregation prevents the server from accessing individual model updates, while differential privacy can be applied to local updates to limit information leakage. In addition, anomaly detection or trust-aware weighting can be incorporated into the adaptive-aggregation process to reduce the impact of malicious or unreliable agents. These extensions are complementary to the proposed framework and represent promising directions for enhancing the privacy and security guarantees of SFDRL.

While the experimental evaluation of SFDRL is conducted in controlled simulated environments, these settings are widely adopted for studying federated reinforcement learning due to their reproducibility and flexibility. Nevertheless, real-world deployment introduces additional challenges, such as unreliable communication, heterogeneous hardware capabilities, delayed updates and non-stationary dynamics.

7.6 Ablation Study

To evaluate the contributions of the key components of SFDRL, we conducted an ablation study by removing one component at a time:

- SFDRL w/o Adaptive Aggregation: All participating agent updates are equally weighted, ignoring performance and stability.
- SFDRL w/o Selective Participation: All selected agents transmit updates regardless of improvement, increasing communication overhead.

Figure 6 summarizes the cumulative reward, convergence speed (number of communication rounds to reach 90% of maximum reward) and total communication cost for each variant.

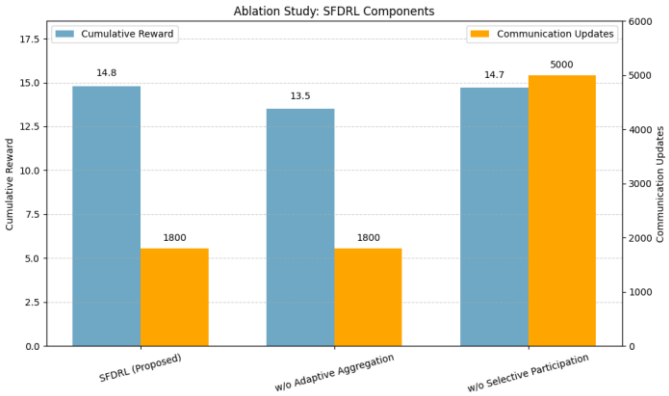

Figure 6. Ablation study results for SFDRL components.

7.7 Discussion

- Removing the adaptive aggregation reduces cumulative reward and slows convergence, indicating that weighting agent contributions according to performance and stability is crucial for effective global learning.
- Removing selective participation increases communication overhead drastically (from 1800 to 5000 updates) with negligible improvement in reward, confirming its role in reducing network load while preserving learning efficiency.
- Together, these results demonstrate that both components are essential for achieving a scalable, stable and communication-efficient federated reinforcement-learning system.

8. CONCLUSION AND FUTURE WORK

In this paper, we proposed a Scalable Federated Deep Reinforcement Learning (SFDRL) framework for collaborative multi-agent learning in heterogeneous and distributed environments. SFDRL integrates adaptive aggregation and selective participation to improve stability, convergence and communication efficiency, achieving near-centralized DRL performance while preserving data privacy. Ablation studies confirmed the importance of both adaptive aggregation and selective participation for effective learning. For future work, we plan to explore more sophisticated aggregation strategies, such as uncertainty-based or trust-aware weighting, extend SFDRL to real-world large-scale applications, like intelligent traffic management and IoT systems, incorporate multi-objective optimization to balance performance, energy efficiency and fairness and investigate advanced privacy-preserving techniques such as differential privacy and secure multiparty computation. Overall, SFDRL provides a scalable, stable and communication-efficient framework that bridges the gap between centralized performance and decentralized, privacy-preserving deployment in collaborative multi-agent systems.

REFERENCES

[1] J. Schulman, F. Wolski, P. Dhariwal, A. Radford and O. Klimov, "Proximal Policy Optimization Algorithms," arXiv preprint, arXiv: 1707.06347, 2017.

[2] V. Mnih et al., "Human-level Control through Deep Reinforcement Learning," Nature, vol. 518, no. 7540, pp. 529-533, Feb. 2015.

[3] L. Li, Y. Lv and F. Wang, "Traffic Signal Timing *via* Deep Reinforcement Learning," IEEE/CAA Journal of Automatica Sinica, vol. 3, no. 3, pp. 247-254, Jul. 2016.

[4] D. Silver et al., "Mastering the Game of Go with Deep Neural Networks and Tree Search," Nature, vol. 529, pp. 484-489, Jan. 2016.

[5] B. McMahan et al., "Communication-efficient Learning of Deep Networks from Decentralized Data," Proc. of the 20th Int. Conf. on Artificial Intelligence and Statistics (AISTATS), vol. 54, pp. 1273-1282, Fort Lauderdale, Florida, USA, 2017.

[6] J. Qi, Q. Zhou, L. Lei and K. Zheng, "Federated Reinforcement Learning: Techniques, Applications and Open Challenges," Intelligence & Robotics, OAE Publishing Inc., DOI: 10.20517/ir.2021.02, 2021.

[7] T. Li, A. K. Sahu, A. Talwalkar and V. Smith, "Federated Learning: Challenges, Methods and Future Directions," IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50-60, May 2020.

[8] Q. Yang, Y. Liu, T. Chen and Y. Tong, "Federated Machine Learning: Concept and Applications," ACM Trans. Intell. Syst. Technol., vol. 10, no. 2, pp. 1-19, Feb. 2019.

[9] W. Zhang et al., "Optimizing Federated Learning in Distributed Industrial IoT: A Multi-agent Approach," IEEE J. Sel. Areas Commun., vol. 39, no. 12, pp. 3688-3703, 2021.

[10] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos and Y. Khazaeni, "Federated Learning with Matched Averaging," arXiv preprint, arXiv: 2002.06440, 2020.

[11] Z. Pan et al., "RFCSC: Communication Efficient Reinforcement Federated Learning with Dynamic Client Selection and Adaptive Gradient Compression," Neurocomputing, vol. 612, p. 128672, 2025.

[12] E. C. Pinto Neto et al., "Federated Reinforcement Learning in IoT: Applications, Opportunities and Open Challenges," Applied Sciences, vol. 13, no. 11, p. 6497, 2023.

[13] Y. Di et al., "FedRL: A Reinforcement Learning Federated Recommender System for Efficient Communication Using Reinforcement Selector and Hypernet Generator," ACM Trans. Recomm. Syst., vol. 4, no. 1, pp. 1-31, 2025.

[14] X. Li, L. Lu, W. Ni, A. Jamalipour, D. Zhang and H. Du, "Federated Multi-agent Deep Reinforcement Learning for Resource Allocation of Vehicle-to-vehicle Communications," IEEE Trans. Veh. Technol., vol. 71, no. 8, pp. 8810-8824, 2022.

[15] N. Zhao et al., "Multi-agent Deep Reinforcement Learning Based Incentive Mechanism for Multi-task Federated Edge Learning," IEEE Trans. Veh. Technol., vol. 72, no. 10, pp. 13530-13535, 2023.

[16] S. Truex et al., "A Hybrid Approach to Privacy-preserving Federated Learning," Proc. of the 12th ACM Workshop on Artificial Intelligence and Security (AISec), pp. 1-11, Nov. 2019.
[17] T. A. Haddad, D. Hedjazi and S. Aouag, "A Deep Reinforcement Learning-based Cooperative Approach for Multi-intersection Traffic Signal Control," Engineering Applications of Artificial Intelligence, vol. 114, p. 105019, 2022.
[18] T. A. Haddad, "Traffic Signal Control for Large-scale Scenario: A Deep Reinforcement Learning-based Cooperative Approach," Proc. of the 12th Int. Conf. Systems and Control (ICSC), pp. 412-417, Batna, Algeria, Nov. 2024.
[19] H. van Hasselt, A. Guez and D. Silver, "Deep Reinforcement Learning with Double Q-learning," Proc. of the 30th AAAI Conf. on Artificial Intelligence (AAAI'16), pp. 2094-2100, 2016.
[20] K. M. Lee et al., "Investigation of Independent Reinforcement Learning Algorithms in Multi-agent Environments," Frontiers in Artificial Intelligence, vol. 5, p. 805823, 2022.
[21] Y. Di et al., "Federated Recommender System Based on Diffusion Augmentation and Guided Denoising," ACM Trans. on Information Systems, vol. 43, no. 2, pp. 1-36, 2025.
[22] A. Tian et al., "Efficient Federated DRL-based Cooperative Caching for Mobile Edge Networks," IEEE Trans. on Network and Service Management, vol. 20, no. 1, pp. 246-260, 2022.
[23] E. Abedini and M. Nickray, "CUBIC-LEARN: A Reinforcement Learning Approach to CUBIC Congestion Control," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 11, no. 4, pp. 466-483, DOI: 10.5455/jjcit.71-1748057293, Dec. 2025.

ملخص البحث:

يمكّن التّعلُّم الاتّحادي من تدريب نماذج التّعلُّم التّعاوني دون مشاركة البيانات الأولية، بينما يوفر التّعلُّم التّعزيزي العميق آليات قوية لاتّخاذ القرارات المتسلسلة. ومع ذلك، يعاني تكاملها من محدودية القابلية للتّوسُّع والحساسية للبيانات غير متطابقة التّوزيع إلى جانب عدم استقرار التّقارب في البيئات الموزّعة. وتقترح هذه الورقة بِنية تعلُّم تعزيزي عميق اتّحادية قابلة للتّوسُّع، حيث تتعلّم العوامل الموزّعة سياساتٍ محلية وتساهم دورياً في نموذج عالمي من خلال استراتيجية تجميع تكيفية تراعي الأداء على العكس من الطّرق التّقليدية التي تعتمد على المتّوسط الموحّد. وتقوم البِنية المقترحة على ترجيح التّحديثات المحلية وفقاً لفعالية تعلّمها، الأمر الّذي يؤدي إلى تقاربٍ أسرع واستقرار أفضل في ظلّ توزيعات البيانات غير المتجانسة، إضافة إلى ذلك، تمّ تقديم آلية اتّصال انتقائية؛ من أجل تقليل عبء الاتّصال بنسبة معتبرة مقارنة بالطُّرق الأخرى.

وقد بينت التّجارب المكثّفة أنّ الطّريقة المقترحة تتفوّق على غيرها من الطّرق، محققةً مكآفاتٍ تراكمية أعلى وتبايناً أقلّ في أثناء التّدريب مع تحسُّنٍ في قابلية التّوسُّع في البيئات الموزّعة كبيرة الحجم. وتؤكد هذه النتائج مُناسبة النّظام المقترح في هذه الدّراسة للتّوظيف العملي في البيئات الذّكية الموزّعة.