

# 无线联邦学习系统自适应延迟等待时间调整算法<sup>①</sup>



张宇<sup>1</sup>, 赵长名<sup>1</sup>, 黄科荣<sup>1</sup>, 李善超<sup>2</sup>

<sup>1</sup>(成都信息工程大学 计算机学院, 成都 610225)

<sup>2</sup>(北德克萨斯大学 计算机科学与工程学院, 登顿 76207)

通信作者: 赵长名, E-mail: [zcm84@cuit.edu.cn](mailto:zcm84@cuit.edu.cn)

**摘要:** 在智慧城市中, 无线通信网络连接了众多边缘智能设备, 用于收集和传输各种类型的数据。为了保护数据隐私, 联邦学习被广泛应用于智慧城市环境中。然而, 城市中的高密度建筑物和复杂多径传播会导致信道衰落, 严重影响通信质量和联邦学习的性能。针对这一问题, 提出了一种基于强化学习的无线联邦学习系统自适应延迟等待时间调整算法。该算法通过强化学习方法自动调整等待时间, 以适应动态变化的信道条件, 缓解由信道衰落引起的数据传输延迟问题, 并通过优化等待时间避免因部分客户端延迟掉队而导致的模型不可信问题。在 MNIST 和 CIFAR-10 数据集上的仿真结果表明, 所提出的算法在保证系统性能的同时能够确保结果的可信性, 并且在多个关键性能指标上显著优于其他基线算法。

**关键词:** 无线联邦学习; 信道衰落; 强化学习; 延迟自适应控制

引用格式: 张宇, 赵长名, 黄科荣, 李善超. 无线联邦学习系统自适应延迟等待时间调整算法. 计算机系统应用, 2025, 34(9): 31-45. <http://www.c-s-a.org.cn/1003-3254/9928.html>

## Adaptive Delay Wait Time Tuning Algorithm for Wireless Federated Learning Systems

ZHANG Yu<sup>1</sup>, ZHAO Chang-Ming<sup>1</sup>, HUANG Ke-Rong<sup>1</sup>, LI Shan-Chao<sup>2</sup>

<sup>1</sup>(School of Computer Science, Chengdu University of Information Technology, Chengdu 610225, China)

<sup>2</sup>(Department of Computer Science and Engineering, University of North Texas, Denton 76207, USA)

**Abstract:** In smart cities, wireless communication networks are employed to connect numerous edge devices for the collection and transmission of various types of data. To safe guard data privacy, federated learning is widely adopted in such urban environments. However, densely packed buildings and complex multipath propagation often lead to channel fading, which significantly degrades communication quality and the performance of federated learning. To address this challenge, a reinforcement learning-based adaptive waiting time adjustment algorithm is proposed for wireless federated learning systems. The waiting time is automatically adjusted using reinforcement learning techniques to accommodate dynamically changing channel conditions. Data transmission delays caused by channel fading are mitigated, and the risk of untrustworthy models due to delayed or dropped clients is reduced by optimizing the waiting time. Simulation results on the MNIST and CIFAR-10 datasets demonstrate that, while ensuring overall system performance, the proposed approach maintains high result fidelity and significantly outperforms baseline algorithms across multiple key performance metrics.

**Key words:** wireless federated learning (WFL); channel fading; reinforcement learning (RL); adaptive delay adjustment

随着无线通讯技术的快速发展及物联网 (Internet of Things, IoT) 和移动边缘设备的普及, 通过无线网络

互联互通的边缘智能设备能够实时采集和传输大量数据, 这些设备和数据正在形成一个庞大的数据生态系

① 基金项目: 四川省科技计划 (2023YFG0301, 2023YFG0018)

收稿时间: 2025-01-13; 修改时间: 2025-01-26; 采用时间: 2025-02-24; csa 在线出版时间: 2025-07-14

CNKI 网络首发时间: 2025-07-15

统,而分布式机器学习也面临前所未有的机遇与挑战.传统的集中式机器学习方法需要将所有数据收集到中央服务器进行训练,这不仅会带来巨额通信开销,还会导致数据隐私泄露<sup>[1]</sup>.为了应对这些挑战,Google在2016年首次提出了联邦学习(federated learning, FL)<sup>[2]</sup>的概念,这是一种允许分布式设备在保护数据隐私的前提下协作训练机器学习模型的范式.FL是一种分布式机器学习方法,其核心思想是让参与联邦训练的客户端在本地完成模型训练,然后将本地模型发送给中央服务器聚合.整个过程中,客户端只需要与中央服务器交换模型参数,无需上传原始数据,这种方式既保护了数据隐私,又降低了通信开销.

这种分布式范式与无线网络具有天然契合性.首先,无线网络中的终端设备通常具备一定的计算和数据存储能力,这为本地模型训练提供了硬件基础.其次,现代无线通信技术的高带宽、低时延特性能够支持模型参数的高效传输.最重要的是,FL通过仅传输模型参数的方式大大减少了通信开销,这对于本就面临带宽受限的无线网络来说尤其重要<sup>[3]</sup>,还能够减轻设备异构对整体学习性能的影响<sup>[4]</sup>,同时满足5G和6G网络的多样化需求.基于这些优势,FL已成为无线网络中实现协作智能的主要技术范式之一.

无线联邦学习(wireless federated learning, WFL)在车联网智能交通、工控自动化以及智慧医疗等多个领域展现出广阔的应用前景.然而,在这些应用场景中,WFL面临诸多挑战,包括通信资源受限、信道衰落、设备异构性、数据分布不均衡及安全隐私威胁等问题<sup>[5]</sup>.首先,由于无线网络中设备的计算能力、存储容量和通信带宽存在显著差异,WFL的训练过程难以保持一致性.部分设备会因算力不足导致本地模型训练速度较慢,而带宽受限的设备则难以及时上传模型参数,从而影响全局模型的聚合效率.其次,WFL中的数据通常具有非独立同分布(non-independent and identically distributed, non-IID)特性,即不同客户端的数据分布存在显著差异.这种情况会导致局部模型难以有效泛化,使全局模型在不同场景下的适应性降低.同时,由于客户端数量庞大且动态加入或退出网络,部分设备会因为网络环境变化无法持续参与训练,导致全局模型的更新周期不稳定.安全性和隐私保护也是WFL面临的重要挑战.无线环境中的恶意攻击(如拜占庭攻击<sup>[6]</sup>、投毒攻击等)会导致模型污染,降低全局模型的可信

度.虽然WFL在一定程度上避免了数据集中存储的隐私风险,但仍然能够通过差分攻击或推理攻击窃取敏感信息.最后,信道衰落是影响WFL稳定性和通信效率的关键因素之一.由于无线信道受限于多径传播、阴影效应、路径损耗等复杂物理环境<sup>[7]</sup>,客户端与服务端之间的信号质量会在不同时间和位置发生剧烈波动.这种动态变化会导致训练过程中数据传输时延增加,甚至发生数据丢失,使部分客户端无法及时上传模型更新,进而影响全局训练的稳定性和收敛性.

为了应对上述挑战,研究人员提出了多种优化方案:在通信方面,采用了压缩量化、重要性采样等技术来减少通信开销<sup>[8]</sup>;在计算方面,提出了设备选择和资源分配策略来平衡系统效率<sup>[9]</sup>;在安全方面则结合差分隐私<sup>[10]</sup>、同态加密<sup>[11]</sup>等技术来增强模型训练的安全性.然而,这些方法在面对无线信道的动态变化时仍然存在局限性,尤其是在全局模型聚合过程中如何动态调整系统延迟等待时间来保证客户端参与度,以应对复杂无线环境带来的训练不稳定性问题,尚未存在有效解决方案<sup>[12,13]</sup>.

强化学习(reinforcement learning, RL)因其能够通过智能体与环境的持续交互来学习最优决策策略的特性,以及在处理动态变化环境下的决策优化问题时表现出的优越性能,已被广泛应用于WFL中解决资源分配、客户端选择<sup>[14]</sup>和通信策略优化<sup>[15]</sup>等问题.其自适应优化能力使其在应对异构网络环境、动态通信条件下的决策优化方面展现出良好的适应性.然而,现有的RL方法主要关注资源分配和设备选择等优化目标,尚未有效解决因无线信道衰落引起的梯度上传不稳定性问题,特别是在客户端随机参与训练的情况下,如何确保全局模型的可信性仍是一个亟待研究的挑战<sup>[16]</sup>.

针对这一问题,本文提出了一种系统级自适应延迟等待时间调整算法,旨在应对无线网络环境的动态变化,提高WFL系统的训练效率和全局模型的可信性.当前的WFL系统通常采用固定的通信时隙或预设的超时机制来控制客户端的梯度上传过程,但这无法适应不断变化的无线信道条件,会导致部分高质量客户端因网络波动错过梯度上传机会,从而影响全局模型的收敛性和稳定性.本文利用双深度Q学习(double deep Q-learning, DDQL)设计了一种自适应延迟等待时间调整机制,使服务器能够根据当前通信环境的状态,智能调整等待时间,最大化接收客户端上传的梯度更

新,同时减少通信延迟对系统性能的影响.本文的主要贡献如下.

(1) 量化了无线通信信道衰落对客户端梯度上传的延迟影响,明确了中心聚合服务器等待时间的动态变化在训练效率、结果准确性和客户端公平性等方面所带来的影响,并基于此构建考虑以上因素的奖励函数.

(2) 设计了 DDQL 算法,用于在联邦学习训练过程中实时调整中心聚合服务器的延迟等待时间,提高客户端参与度并减少系统性能损耗.

(3) 对比了 DDQL 与其他基准 RL 算法,结果表明 DDQL 在模型准确性、延迟以及客户端公平性方面均具有显著优势.

(4) 所提算法可广泛应用于车联网、工业物联网和智慧医疗等对训练可靠性要求较高的领域,为在复杂无线环境中实现鲁棒的 WFL 优化提供了新的思路.

## 1 相关工作

### 1.1 无线联邦学习

WFL 是将 FL 与无线通信技术结合起来的一种新兴研究方向.传统的联邦学习旨在保护用户数据隐私,通过将模型更新发送到中央服务器而不是共享原始数据,来完成跨设备的机器学习任务.在无线环境下,WFL 需要考虑通信延迟、带宽限制以及数据传输的可靠性等问题.

压缩算法在 WFL 中用于减少每轮通信中传输的数据量,以应对有限带宽的挑战.压缩方法通常包括参数量化、梯度压缩等,能够显著降低网络开销.Liu 等人<sup>[17]</sup>探讨了如何通过量化技术压缩梯度,以减少传输的模型参数量,其提出了一种基于量化的分布式训练方法,在带宽受限的环境中有效地提高了训练效率.文献<sup>[18]</sup>提出了基于投影的稀疏三元压缩算法,通过在客户端和服务端进行梯度压缩,降低通信成本,并在服务端采用梯度投影的聚合策略,以缓解客户端数据非独立同分布导致的不利影响.

在传统 WFL 中,客户端和服务端通常需要同步更新,等待每个客户端上传模型更新后,服务器再进行全局聚合.异步更新方法<sup>[19]</sup>允许客户端根据自己的训练进度进行更新,而无需等待其他客户端,不仅可以减少同步等待时间,还能够提高训练效率.文献<sup>[20]</sup>提出了一种异步联邦学习框架,允许客户端以不同步的方式进行更新.该方法适用于带宽受限或网络波动较大的

情况,能够显著减少等待时间和网络开销.

模型精简技术通常用于减少模型的大小和计算复杂度,这在无线传输带宽受限或计算资源有限的情况下尤其重要.精简方法包括剪枝、权重共享和低秩矩阵分解等.Tao 等人<sup>[21]</sup>提出了在 FL 中使用模型剪枝的方法.通过剪枝来降低模型的复杂度,从而减少通信量,使其适用于 WFL 环境.文献<sup>[22]</sup>提出了一种通过去除冗余数据来提升联邦学习通信效率的方法.其分析了冗余更新参数的根源,结合数据非独立同分布特性与分布式模型训练过程,定义了核心数据集敏感度与损失函数容忍度,从而构建联邦核心数据集.随后,又设计了分布式自适应模型演化机制,使得在每轮训练迭代前都可动态调整模型的结构和大小,在减少终端与服务端间传输数据量的同时,保证训练模型准确率.

WFL 由于网络波动和带宽限制,客户端之间的通信效率大不相同,因此采用适应性优化算法来动态调整学习率和更新策略是提高系统整体效率的重要方法.文献<sup>[23]</sup>提出了一种适应性联邦学习算法,旨在处理无线网络中的不稳定通信问题,通过动态调整学习率和更新频率,能够在带宽波动的情况下有效优化训练.

针对无线网络的频繁波动,研究者们提出了一些智能调度策略,以保证 WFL 的高效通信和低延迟.Chen 等人<sup>[24]</sup>提出了一种优化的无线网络调度算法,旨在通过调度策略减少无线网络中的传输延迟和带宽占用.研究表明,即使在波动较大的网络环境中,优化调度也能够有效提高 WFL 的性能.Shaban 等人<sup>[25]</sup>分析了脉冲噪声对 WFL 系统的影响,其提出 Sk-OTAF 方法可以保证客户端能够在脉冲噪声波动下有效地参与联邦学习.

### 1.2 强化学习在无线联邦学习中的应用

在 WFL 中,RL 的应用正在成为一个新的研究热点.在面对不稳定的网络环境、资源约束、异质客户端等挑战时,RL 可以在 WFL 的背景下发挥重要作用.RL 通过智能决策来优化系统的资源分配、通信策略、学习过程等方面,可以有效提升 WFL 的效率和稳定性.

在 WFL 中,客户端资源(如计算能力、网络带宽)差异较大,这种异构环境对训练稳定性造成了很大影响.RL 可以通过智能的客户端选择策略来提高训练的效率.文献<sup>[26]</sup>提出了 FLASH-RL 框架,利用 DDQL 来解决 FL 中的系统异质性和静态异质性问题.其引入了一种基于声誉的效用函数,通过评估客户端的当前和

过去表现来衡量其贡献,从而完成动态客户端选择和任务分配。

强化学习可以用于智能决策,以优化通信策略,减少网络开销,并提高训练效率。文献[27]利用深度强化学习(deep reinforcement learning, DRL)对无线联邦学习中的资源调度问题进行优化,通过构建一个智能代理,动态调整客户端的上传频率和数据量,提高了训练效率并减少了通信开销。

Zhang 等人[28]提出了一种基于移动边缘计算的联邦学习系统,其将客户端选择问题建模为马尔可夫决策过程,通过强化学习方法自适应选择每轮通信中的参与设备。这种选择机制无需预先了解环境信息,能够在保证设备参与度的同时优化系统性能。Mao 等人[29]针对无线联邦学习网络中通信资源受限和用户设备异构性问题,提出了一种基于深度强化学习的联合客户端选择与带宽分配框架 CSBWA,其根据客户端状态、历史带宽使用情况和反馈奖励自动执行策略决策,有效降低了时间开销和能量消耗,同时保证了模型性能。

相较于传统的 RL 方法,自适应强化学习(adaptive reinforcement learning, ARL)能够根据不断变化的无线环境和系统状态实时调整策略,从而增强 WFL 的稳定性和鲁棒性。在动态信道环境下,ARL 可用于自适应通信优化,确保在信道衰落、网络拥塞等不确定因素影响下,客户端仍能有效上传模型更新。例如,文献[30]提出了一种基于自适应强化学习的动态通信调度方法,通过结合部分可观测马尔可夫决策过程,智能调整每轮通信的客户端选择和传输功率,以降低通信成本并减少丢包率。此外,文献[31]采用多智能体深度强化学习(multi-agent deep reinforcement learning, MADRL)方案,使得客户端能够在分布式协作环境中自主学习最优通信策略,在不依赖中心服务器全局信息的情况下,实现更加灵活的通信资源管理。在设备异构性方面,ARL 可用于动态任务分配与模型压缩,以应对计算资源和带宽受限的问题。文献[32]提出一种基于自适应强化学习的异构设备感知训练框架,利用自适应神经架构搜索(neural architecture search, NAS)技术,根据客户端计算能力与存储能力自动调整模型的复杂度,在保证模型收敛性的同时减少计算和通信负担。ARL 在 WFL 系统中的应用涵盖了通信优化、异构设备管理、任务分配、梯度压缩和动态客户端调度等多个方面,但是缺少一种系统级自适应延迟等待时间调整方法[33]。

## 2 自适应延迟等待时间调整算法设计

本节针对信道衰落导致的系统时延问题和可靠性问题,设计了一个动态调整 WFL 系统延迟等待时间的算法框架。首先,结合 Rayleigh 信道衰落模型[34]与无线传输模型[35],对信道衰落引起的延迟特性进行量化分析,并构建客户端参与度的量化评估框架,用于评估全局模型聚合结果的可信性。然后,基于 DDQL 提出了一种自适应延迟等待时间调整算法。该算法的奖励函数在两个相互制约的因素之间进行权衡:延长延迟等待时间可提升客户端参与度,从而提高结果可信度;但过长的等待时间会导致系统性能下降并增加资源消耗。因此,该算法旨在确定每轮训练的最优延迟等待时间调整策略。本节将详细介绍各关键模块的具体设计与实现。

### 2.1 基于客户端参与度的结果可信性量化

为了量化联邦学习结果的可信性,本文将其转化为联邦学习用户的成功参与比例问题。当成功参与训练的客户端比例达到预定阈值时,WFL 的结果满足可信性要求[29]。基于此,本文以成功传输并参与聚合的用户比例作为衡量可信性的度量指标,具体定义如下:

$$Success\_Rate = \frac{num_{fulfil}}{num_{all}} \quad (1)$$

其中,  $num_{all}$  代表参与 WFL 的客户端总数,  $num_{fulfil}$  代表满足延迟要求的客户端数量,  $Success\_Rate$  代表能够成功上传梯度并参与聚合的客户端比例。

### 2.2 信道衰落环境下的时延分析与建模

在 WFL 环境中,密集建筑物分布、多径传播效应以及信道的随机波动都会导致信道衰落,对无线通信造成干扰,引发传输时延的增加和模型聚合速率的降低。受信道衰落影响较大的客户端在预定时间内无法完成梯度上传,这对联邦学习结果的准确性会造成一定影响。

本节基于正交频分多址(OFDMA)技术构建传输模型[35],通过分配资源块实现客户端与中央服务器之间的高效数据上传。针对无线信道的动态特性,引入 Rayleigh 衰落模型描述信道增益的随机变化,以刻画多径传播效应对传输性能的影响。在此基础上,结合 Shannon 定理计算了系统的平均传输速率,并基于该速率进一步分析了数据上传过程中的延迟特性,进而提出了一种面向信道衰落的时延计算方法。

### 2.2.1 传输模型

在 WFL 系统中, 中央服务器每轮选择固定比例的用户参与训练, 根据文献[35]中定义的传输模型, 在上行链路中, 本文假设系统采用正交频分多址 (OFDMA) 技术, 其中每个用户平均分配资源块. 假设系统共有  $R$  个资源块, 每轮选中  $K$  个用户参与训练, 则被选中的每个用户在每轮训练中平均分配得到的资源块数量为  $R/K$ . 在此基础上, 用户  $i$  向基站上传其本地联邦学习模型参数的上行传输速率可以表示为式 (2).

$$C_i(P_i) = \frac{R}{K} B \mathbb{E}_{h_i} \left( \log_2 \left( 1 + \frac{P_i h_i}{I + BN} \right) \right) \quad (2)$$

其中,  $R/K$  表示每个用户分配的资源块数量,  $B$  表示每个资源块的带宽,  $h_i$  表示用户  $i$  与基站之间的信道增益,  $P_i$  表示用户  $i$  的发射功率,  $\mathbb{E}_{h_i}(\cdot)$  表示对信道增益  $h_i$  的期望,  $N$  表示噪声功率谱密度,  $I$  表示由其他服务区域用户共享资源块所引入的干扰功率. 通过假设每轮训练中用户平均分配资源块, 简化了资源分配的建模复杂度, 保证了 WFL 系统中每个用户在上传联邦学习模型参数时具有相等的资源分配机会, 同时考虑了信道衰落和系统噪声及干扰的影响, 以模拟真实的无线通信环境下 WFL 的传输速率.

### 2.2.2 Rayleigh 衰落模型

在 WFL 中, 信道衰落是影响信号传播质量的关键因素. 本文采用 Rayleigh 衰落模型<sup>[34]</sup>对信道增益进行建模, 定义如下:

$$h_i = o_i d_i^{-2} \quad (3)$$

其中,  $h_i$  表示用户  $i$  与基站之间的信道增益,  $o_i$  是 Rayleigh 衰落因子, 通常假设为一个服从 Rayleigh 分布的随机变量, 用来描述快速信道衰落的影响,  $d_i$  表示用户  $i$  与基站之间的距离. Rayleigh 衰落因子是一种快速衰落特性, 用来建模无线信号在多径传播环境下的变化, 由于信号传播时会受到建筑物、地形等障碍物的反射、折射和散射, 导致信号在 WFL 服务器接收端的幅度和相位发生快速变化. 式 (3) 综合考虑了路径损耗 (慢衰落) 和 Rayleigh 衰落 (快衰落) 的共同作用, 可以更准确地刻画信道增益特性, 描述用户间由于距离和信道环境差异而导致的通信质量差异.

### 2.2.3 延迟计算

本文假设客户端数据分布为独立同分布 (IID), 且忽略客户端计算能力的差异对延迟的影响. 研究重点

集中在通信过程中由信道条件引发的传输延迟问题, 将传输延迟作为处理信道干扰与信道衰落的优化目标函数进行建模和分析. 传输延迟是指 WFL 过程中客户端上传本地模型参数至中央服务器所需的时间, 其大小直接影响到 WFL 系统的整体性能.

为了简化问题, 本文仅考虑通信链路中的传输时间, 而忽略其他潜在的计算和调度延迟. 最小化传输延迟可以通过优化通信链路条件来实现, 包括信道分配、发射功率以及干扰控制等因素. 本文从 WFL 系统中心服务器控制角度出发, 通过动态调整延迟等待时间以接收大部分用户的梯度更新, 以此抵消信道衰落对系统的影响, 提高 WFL 的结果准确性和公平性. 在理想条件下传输延迟计算方法如下:

$$T_i^{\text{transmission}_k} = \frac{\sigma(m_k)}{B \log_2 \left( 1 + \frac{P_i o_i d_i^{-2}}{I + BN} \right)} \quad (4)$$

其中,  $T_i^{\text{transmission}_k}$  表示用户  $i$  传输模型  $k$  所需要的时间,  $\sigma(m_k)$  表示用户维护的本地模型  $m_k$  的大小, 其余参数上文已做介绍. 根据式 (4) 可以计算出 WFL 系统中每个用户上传梯度到中心聚合服务器的传输延迟时间.

### 2.3 基于 DDQL 的自适应延迟等待时间调整算法

图 1 为本文提出的基于 DDQL 的自适应延迟等待时间控制系统架构. 系统由多个客户端的本地训练模块和全局聚合模块组成, 服务器在每轮训练时都会收集当前网络环境状态, 包括信道质量和客户端参与情况等, 并将其输入给 DDQL 智能体进行决策. DDQL 根据经验回放不断更新 Q 值网络, 在获得最优动作后动态调整服务器的等待时间, 从而在异构无线环境下实现对传输时延的自适应调控. 通过这一不断迭代的交互与调控机制, WFL 系统能够在保证高效模型聚合的同时, 最大程度地减少因信道衰落等网络不稳定因素所带来的训练延迟.

系统流程图如图 2 所示. 首先, WFL 系统完成初始化并向所有参与用户分发模型, 让其进行预训练; 中心聚合服务器记录各用户的传输时延, 并以平均时延作为初始等待时间. 随后, 每轮从用户集随机抽取一定比例的用户参与训练, 服务器仅接收符合时延要求的用户上传的梯度, 并同时更新参与用户的传输时延. 接着, DDQL 根据所获得的状态和奖励进行训练, 更新 main network 和 target network, 输出本轮的 best action,

即新的最优延迟等待时间调整动作. 随后, 系统根据此 best action 进行自适应调控, 更新服务器端的等待时间

设置, 并进入下一轮训练. 如此迭代往复, 直至达到预设的训练轮次要求为止.

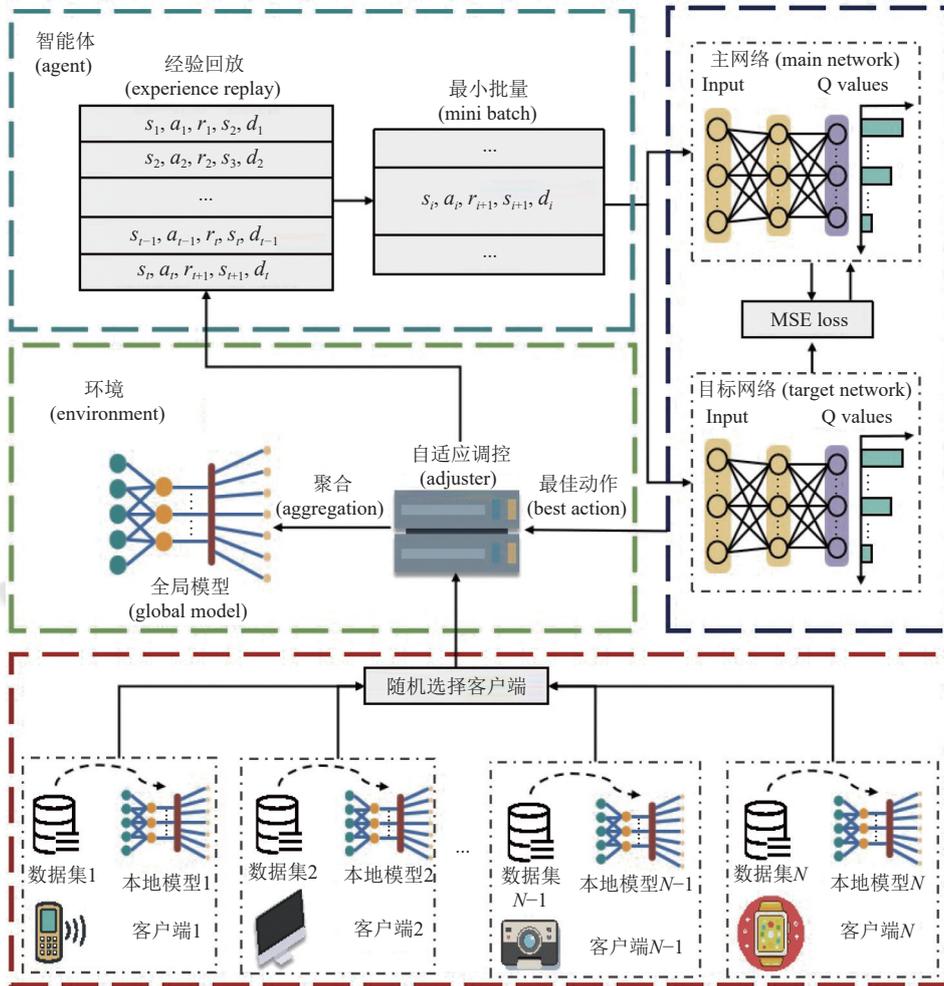


图1 自适应延迟等待时间控制系统架构图

### 2.3.1 马尔可夫决策过程

为了将 RL 应用于自适应控制过程, 需要将其形式化为一个马尔可夫决策过程 (Markov decision process, MDP). 在本文的建模中, 关键元素包括: 状态空间、动作空间和奖励函数.

(1) 状态空间: 本文将状态空间定义为式 (5), 表示当前系统第  $t$  轮训练的延迟等待时间以及能够成功接收到的客户端上传梯度的比例. 随着延迟等待时间的变化, 成功接收客户端的比例也会发生变化. 根据文献[36], 本文的目标是覆盖至少 70% 参加训练的用户, 才可以满足结果可信需求.

$$S_t = (Wait\_time_t, Success\_Rate_t) \quad (5)$$

其中,  $S_t$  表示第  $t$  轮训练时 WFL 系统的状态记录,  $Wait\_$

$time_t$  表示当前系统延迟等待时间,  $Success\_Rate_t$  表示当前可以成功接收梯度更新的客户端比例.

(2) 动作空间: 系统根据当前覆盖的客户端比例动态调整延迟等待时间, 如式 (6) 所示. 调整因子在 0.7-1.3 之间均匀分布, 表示当前延迟时间最多增加 130% 或减少 70%.

$$a_t = \{actions[i], i \sim Uniform(0.7, 1.3)\} \quad (6)$$

(3) 奖励函数: 本文设计的奖励函数由两部分组成. 第 1 部分通过用户覆盖比例的变化, 以量化增加或减少延迟等待时间所带来的结果公平性和准确性评估变化; 第 2 部分则通过刻画延迟等待时间对系统性能的影响, 评估其对整体性能的作用.

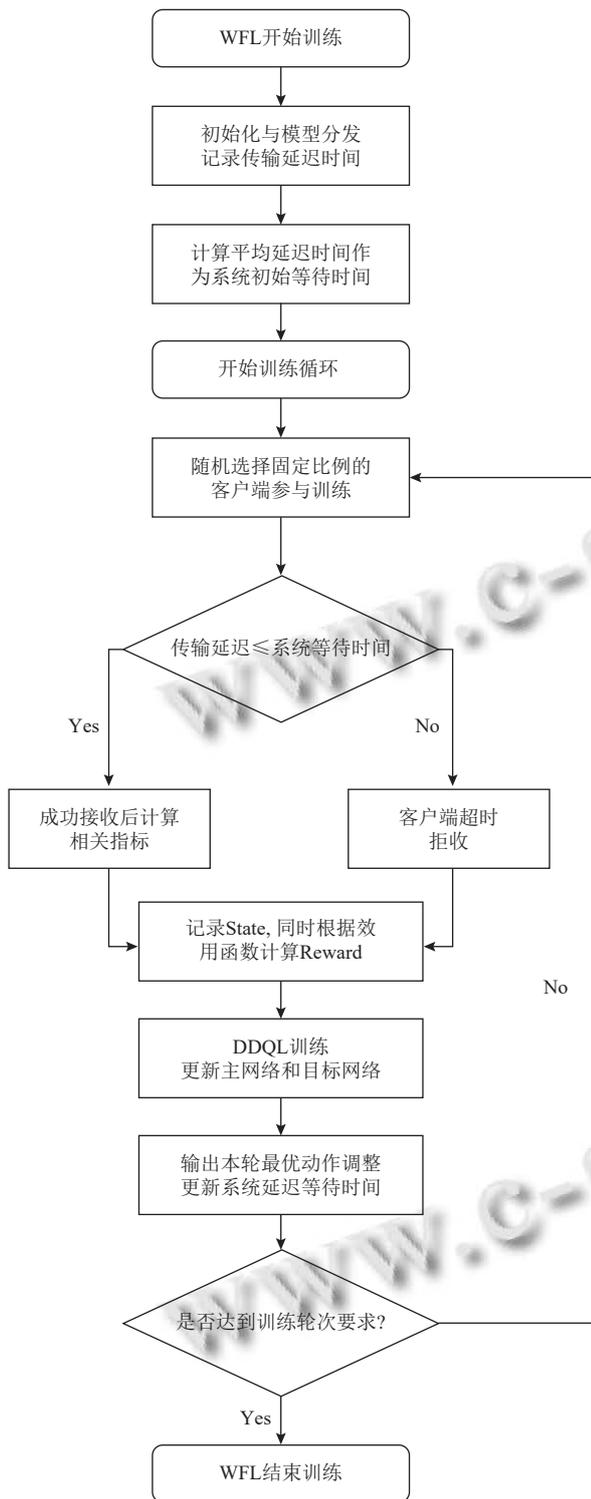


图2 系统流程图

首先,在WFL系统中,所有用户接收全局模型并进行一轮预训练,中心服务器记录每个用户的初始传输延迟时间。随后,设置初始等待时间作为基准,以计算满足延迟要求的客户端数量。通过将满足延迟要求

的客户端数量除以参与训练的客户端总数,得到覆盖比例作为奖励函数的第1部分  $R_1^t = Success\_Rate$ 。其次,为了量化延迟等待时间的变化对系统性能的影响,本文对当前延迟等待时间进行归一化处理,并结合非线性函数  $f(x) = 1 - x^2$  构造奖励的第2部分,如式(7)所示:

$$R_2^t = 1 - \left( \frac{wait\_time_{now} - latency_{min}}{latency_{max} - latency_{min}} \right)^2 \quad (7)$$

其中,  $wait\_time_{now}$  表示的是系统当前的延迟等待时间,  $latency_{min}$  是参与WFL训练的用户最小传输延迟时间,  $latency_{max}$  是参与WFL训练的用户最大传输延迟时间。该函数能够有效模拟系统性能变化:随着延迟等待时间的增加,系统整体性能呈现下降趋势,且下降速率逐渐加快。当系统等待时间满足所有客户端的参与要求时,奖励归0。

本文设计的奖励函数一方面通过用户覆盖比例的变化来衡量联邦训练过程中客户端参与度和结果准确性的平衡,另一方面利用基于系统等待时间的非线性惩罚函数来动态控制延迟对整体性能的影响。通过这两项指标的线性组合,能够在提升模型准确度与客户端参与度的同时,有效抑制过度等待带来的系统性能损耗。奖励函数如式(8)所示,其中  $Reward_t$  表示根据第  $t$  轮动态调整的延迟等待时间进行聚合后获得的奖励,  $\alpha$  表示联邦聚合参与度的重要性权重,  $\beta$  表示维持WFL系统性能的重要性权重。

$$Reward_t = \alpha \times R_1^t + \beta \times R_2^t \quad (8)$$

### 2.3.2 基于DDQL的解决方案

本文引入  $Q(s_t, a_t)$  来表示状态  $s$  下中央服务器的动作选择值。为了选择最大化当前奖励函数的最佳动作,将最高  $Q$  值作为优先考虑的目标。最初,本文鼓励使用  $\epsilon$ -greedy 进行探索,并随着训练轮数的增加逐渐减小  $\epsilon$  值。

本文使用了DDQL算法<sup>[37]</sup>。其中DDQL代理由两个神经网络组成:主网络和目标网络。主网络用于训练,而目标网络评估下一个状态的动作,并每  $P$  步定期更新。此外,DDQL代理还采用了重播记忆机制来打破连续经验  $(s_i, a_i, s_{i+1}, r_{i+1}, d_i)$  和  $(s_{i+1}, a_{i+1}, s_{i+2}, r_{i+2}, d_{i+1})$  之间的相关性,其中  $d_i$  表示一个布尔值,指示最终是否RL问题可以定义为最小化目标值和近似值之间的均方误差(MSE)损失,如式(9)所示:

$$L_t = (Y_t - Q_{\text{main}}(s_t, a_t))^2 \quad (9)$$

其中,  $Y_t$  是动作  $a_t$  在第  $t$  轮的目标值, 如式 (10) 所示:

$$Y = r_t + \gamma \cdot Q_{\text{target}}(s_{t+1}, \arg \max Q_{\text{main}}(s_{t+1}, a_t)) \quad (10)$$

其中,  $r_t$  为每轮训练中心服务器动态调整延迟等待时间的奖励,  $\gamma$  表示未来奖励的权重. 该损失是在第  $t$  轮训练中计算的, 用于由中央服务器动态调整延迟等待时间. 本文研究的重点是根据状态选择最合适的动作. 该算法的流程如算法 1 所示.

算法 1. 基于 DDQL 的延迟等待时间自适应控制算法

输入: 训练总轮数  $MaxEpisodes$ , 每轮最大训练步数  $MaxStepsInEpisode$ , 初始探索率  $\epsilon_{\text{init}}$ , 最小探索率  $\epsilon_{\text{min}}$ , 衰减因子  $decay$ , 批次大小  $Batchsize$ ,  $target\ network$  同步周期  $P$ , 折扣因子  $\gamma$ ,  $ReplayBuffer$  容量 (包含存储状态、动作、奖励和下一状态信息), 状态空间  $S = (\text{Wait\_time}, \text{Success\_Rate})$ , 动作空间  $A = \{\text{actions}[i]\}$ .

输出: 训练得到的主网络参数  $\theta_{\text{main}}$ , 即最优调整策略. 根据各状态下输出的动作实现对延迟等待时间的自适应控制.

```

1. function DDQL_Training
2. 初始化
3. 初始化主网络  $Q_{\text{main}}$  和目标网络  $Q_{\text{target}}$ 
4. 初始化经验回放池  $ReplayMemory\ D$ 
5. 随机初始化主网络参数  $\theta_{\text{main}}$ 
6. 同步到目标网络  $\theta_{\text{target}} \leftarrow \theta_{\text{main}}$ 
7. 初始化  $\epsilon$ -greedy 的探索率  $\epsilon$ 
8. 初始化训练步数计数器  $t \leftarrow 0$ 
9. FOR  $episode=1$  TO  $MaxEpisodes$  DO
10. 设置环境初始状态:
11.  $S_0 = (\text{Wait\_time}_0, \text{Success\_Rate}_0)$ ;
12. FOR  $step=1$  TO  $MaxStepInEpisode$  DO
13. 动作选择:
14.  $a_t = \begin{cases} \text{以概率 } \epsilon \text{ 随机选择动作} \\ \text{以概率 } 1-\epsilon \text{ 随机选取使 } Q_{\text{main}} \text{ 取最大值的动作} \end{cases}$ 
15. 执行动作  $a_t$ , 获取即时奖励  $r_t$  和下一状态  $s_{t+1}$ , 同时标记是否达到终止
16. 将经验数据  $(s_t, a_t, s_{t+1}, r_{t+1}, d_t)$  存入经验回放池  $D$ 
17. IF  $D.size > Batchsize$  THEN
18. 从经验池  $D$  中随机抽取大小为  $Batchsize$  的训练批次
19. FOR each  $(s_i, a_i, s_{i+1}, r_{i+1}, d_i)$  IN  $miniBatch$  DO
20. IF  $d_i = \text{True}$  THEN
21. 目标值  $Y_i \leftarrow r_i$ ;
22. ELSE
23. 令下一状态的最优动作  $a_{\text{next}} \leftarrow \arg \max_a Q_{\text{main}}(s_{i+1}, a)$ ;
24. 计算目标值  $Y_i \leftarrow r_i + \gamma \cdot Q_{\text{target}}(s_{i+1}, a_{\text{next}})$ ;
25. 计算损失函数  $loss_i \leftarrow (Y_i - Q_{\text{main}}(s_i, a_i))^2$ ;
26. 执行梯度下降, 更新主网络参数  $\theta_{\text{main}}$ 
27. 更新训练步数  $t \leftarrow t+1$ ;
28. IF  $(t \bmod P) = 0$  THEN
29. 目标网络参数同步  $\theta_{\text{target}} \leftarrow \theta_{\text{main}}$ ;
30. 更新探索率  $\epsilon \leftarrow \max(\epsilon_{\text{min}}, \epsilon \cdot decay)$ ;
31. IF  $done = \text{True}$  THEN BREAK

```

```

32. END FOR
33. END FOR
34. RETURN  $\theta_{\text{main}}$ 
35. END function

```

算法 1 基于 DDQL, 旨在根据环境状态输出最优动作, 进而自适应地调整延迟等待时间. 该算法首先随机初始化主网络和目标网络参数, 并设置经验回放池、探索率等训练超参; 随后在多轮迭代中, 利用  $\epsilon$ -greedy 策略从主网络中选择动作, 与环境交互以获取奖励和新状态, 再将交互样本存入经验回放池并从中抽取样本来训练网络参数. 若到达终止状态或完成最大步数, 即结束当前轮次; 与此同时, 主网络会周期性地与目标网络同步, 并逐步衰减探索率, 以在后期更好地利用已学得策略. 经过多轮迭代训练, 算法会输出训练好的主网络参数, 用于在实际 WFL 系统中依据状态选取动作, 从而在保证用户参与度和结果准确性的前提下动态控制延迟等待时间.

### 3 仿真实验

在本节中, 为了验证所提出的 WFL 系统自适应延迟等待时间调整算法的有效性, 本文设计了全面的实验模拟环境并开展对比分析. 实验环境模拟了典型的无线通信场景, 其中客户端分布于异构网络中, 具有不同的通信能力. 所使用的数据集包括广泛应用于联邦学习的 MNIST 和 CIFAR-10, 以体现算法在图像分类任务中的表现. 仿真实验中, 关键参数如客户端数量、通信带宽限制、通信能力不均衡程度、通信延迟等均被合理设定, 以确保实验的真实性和全面性.

此外, 中心服务器的聚合算法采用经典的 FedAvg 方法<sup>[2]</sup>, 通过在每一轮中让客户在本地进行多步梯度下降, 并将模型参数平均聚合到中心服务器, 从而实现分布式模型训练, 如式 (11) 所示,  $W_{t+1}$  为聚合后的全局模型参数,  $n_k$  表示客户端  $k$  的数据量,  $w_t^k$  为第  $k$  个客户端在第  $t$  轮训练的本地模型参数.

$$W_{t+1} = \sum_{k=1}^K \frac{n_k}{n} w_t^k \quad (11)$$

最后, 通过与多种主流强化学习算法: Q 学习算法 (Q-learning)<sup>[38]</sup>、深度 Q 网络 (deep Q-network, DQN)<sup>[39]</sup>、近端策略优化算法 (proximal policy optimization, PPO)<sup>[40]</sup> 和最大熵强化学习算法 (soft actor-critic, SAC)<sup>[41]</sup> 进行对比, 全面验证所提出 DDQL 算法在自适应延迟等待

时间调整上的有效性. 性能评价从模型的收敛速度、分类精度、系统公平性以及通信开销等多个维度展开. 实验结果表明, 所提出的 DDQL 算法在异构无线网络环境中表现出显著优势, 不仅可以覆盖更多客户端以保证结果准确性和公平性, 还能够有效控制延迟等待时间的调整幅度, 确保了 WFL 系统的整体性能.

### 3.1 仿真方法

#### 3.1.1 仿真设置

本实验在 Windows 11 专业版操作系统上完成, 采用 Python 3.8.19 编程语言, 深度学习框架为 PyTorch 2.2.2. 硬件平台为 12th Gen Intel(R) Core(TM) i3-12100F 3.30 GHz 处理器, 搭载 NVIDIA RTX 3050 显卡 (12 GB 显存) 及 16 GB DDR4 内存. 为保证结果的可复现性, 随机种子设置为 42.

#### 3.1.2 仿真数据集设置

(1) MNIST 数据集<sup>[42]</sup>: 这是一个用于图像分类任务的手写数字数据集. 数据集包含 60 000 张训练图像和 10 000 张测试图像. 本文在仿真实验中使用了一个卷积神经网络 (CNN) 模型, 该模型包括 2 个  $5 \times 5$  的卷积层. 第 1 层有 20 个输出通道, 第 2 层有 50 个输出通道. 每个卷积层后都接 1 个  $2 \times 2$  的最大池化层.

(2) CIFAR-10 数据集<sup>[43]</sup>: 该数据集由 60 000 张彩色图像组成, 共分为 10 个类别, 每个类别包含 6 000 张图片. 本文训练了一个 CNN 模型, 模型包含 2 个  $5 \times 5$  的卷积层. 第 1 层有 6 个输出通道, 第 2 层有 16 个输出通道. 同样, 每个卷积层后都接 1 个  $2 \times 2$  的最大池化层.

在实验中, 为了对 MNIST 和 CIFAR-10 数据集进行划分, 本文使用了 FedLab 框架<sup>[44]</sup>, 该框架提供了多种数据划分策略. 然而, 为了控制变量并专注于研究通信相关的影响, 本文仅采用 IID 划分. 在这种划分方式下, 数据被随机均匀地划分给每个客户端, 确保每个客户端分配到的样本数量相等, 从而避免了数据分布不均引入的额外变量干扰.

此外, 为模拟无线联邦学习系统中的多用户场景, 本文设置了 100 个用户 (客户端). 每个客户端在 IID 划分下获得数据集的一部分, 从而使实验结果能够更准确地反映通信条件对系统性能的影响. 以 CIFAR-10 数据集划分为例, 前 10 个客户端的数据分布如图 3 所示.

#### 3.1.3 传输信道参数设置

本文在软件层面为参与 WFL 的客户端随机生成

无线信道的关键参数, 包括带宽、发射功率、Rayleigh 衰落因子、噪声等, 并由此计算传输时延来模拟真实的 WFL 环境. 根据文献<sup>[45]</sup>, 表 1 列出了该环境下的传输信道超参数具体设置, 包括带宽、信号发射功率、Rayleigh 衰落因子、传输距离、噪声功率谱密度以及用户间噪声功率等关键参数. 除 Rayleigh 衰落因子服从 Rayleigh 衰落分布外, 其余均符合范围内的均匀分布, 为 100 个用户分别设置传输信道状态.

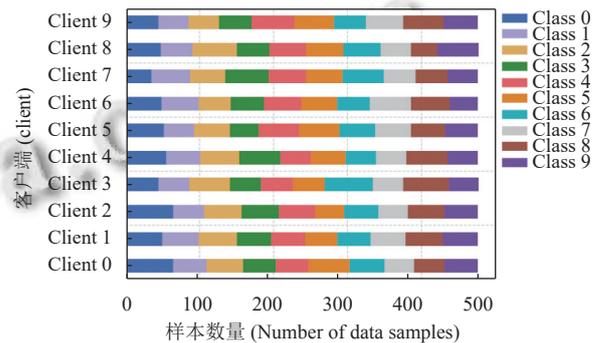


图 3 数据划分结果图

表 1 传输信道超参数设置

环境参数	符号	数值范围
带宽	$B$	0.08–0.12 MHz
信号发射功率	$P_i$	0.001–0.01 W
Rayleigh衰落因子	$\sigma_i$	Rayleigh(1)
传输距离	$d_i$	0.1–1.5 km
噪声功率谱密度	$N$	-174 dBm/Hz
用户间噪声功率	$I$	$10^{-5}$ – $10^{-4}$ W

注: Rayleigh(1)表示Rayleigh衰落因子服从尺度参数为1的Rayleigh衰落分布

#### 3.1.4 仿真实验超参数设置

为了验证本文所提出 DDQL 算法的优越性, 本文选取了 4 种主流强化学习算法作为对比对象, 具体包括 Q-learning<sup>[38]</sup>、DQN<sup>[39]</sup>、PPO<sup>[40]</sup>以及 SAC<sup>[41]</sup>, 这些算法适用于不同的应用场景中, 但是在 WFL 环境中, DDQL 可以展现出优势和适用性. 表 2 列出了这 5 种算法的关键超参数设置, 包括学习率、折扣因子、探索策略、批量大小、记忆库大小以及目标网络更新频率等, 以便于在相同条件下进行公平的性能比较. 更新频率表示每几轮更新一次.

FedAvg 中设置客户端数量为 100, 每轮选择 10% 的客户端参与训练, 全局训练轮数设置 100 轮, 每个被选中的客户端本地训练 3 轮, 客户端本地训练的批大小设置为 32, 本地训练的学习率设置为 0.01, CIFAR-10 和 MNIST 数据集选择 IID 划分.

表2 传输信道超参数设置

超参数	Q-learning	DQN	PPO	SAC	DDQL
学习率	0.1	0.001	3E-4	3E-4	0.001
折扣因子 $\gamma$	0.9	0.99	0.99	0.99	0.99
批次大小	N/A	32	64	256	32
重放内存容量	N/A	10000	100000	100000	10000
更新频率	N/A	100	100	1	1000

### 3.2 仿真结果

本文分别在 MNIST 和 CIFAR-10 基础数据集上进行模拟实验, 根据传输信道超参数, WFL 系统中心服务器首先记录 100 个客户端的传输延迟时间, 时间分布如图 4 所示。

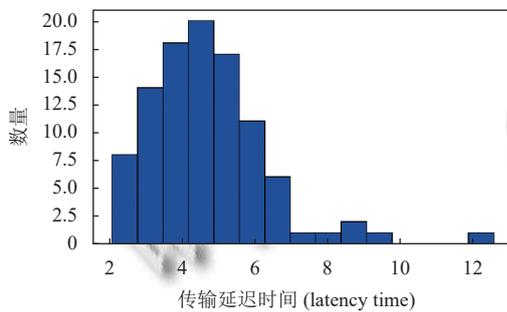


图4 初始延迟等待时间分布

图 5 展示了强化学习模拟结果对比. 从图中可以看出, 每种强化学习算法都在训练过程中不断更新其策略, 对等待时间进行相应的动态调整, 从而在“系统延迟等待时间”和“用户覆盖比例”等指标之间进行平衡. 随着训练轮次的增加, 算法能够自适应地根据环境反馈来改善其策略: 一方面逐步降低不必要的等待时间以提高效率, 另一方面确保足够的客户端能够及时上传模型梯度, 从而维持较高的结果准确性。

图 5 从多个角度进行了算法的对比分析. 与其他基线强化学习的算法相比 (Q-learning、DQN、PPO、SAC), DDQL 更加适合 WFL 系统的延迟等待时间自适应调控场景. 首先, 相较于 DQN, DDQL 通过减少 Q 值的过估计, 使其具有更好的稳定性和收敛性, 特别适合在延迟等待时间调整任务中应用. 其次, DDQL 具有更高的稳定性和较快的实时响应能力, 虽然其计算开销略有提升, 但训练过程更加稳定且收敛速度更快, 并且在移动边缘计算设备可以承受的范围内. 与 PPO 和 SAC 相比, DDQL 不需要很大的算力和存储空间, 完美契合移动边缘计算环境的轻量性要求。

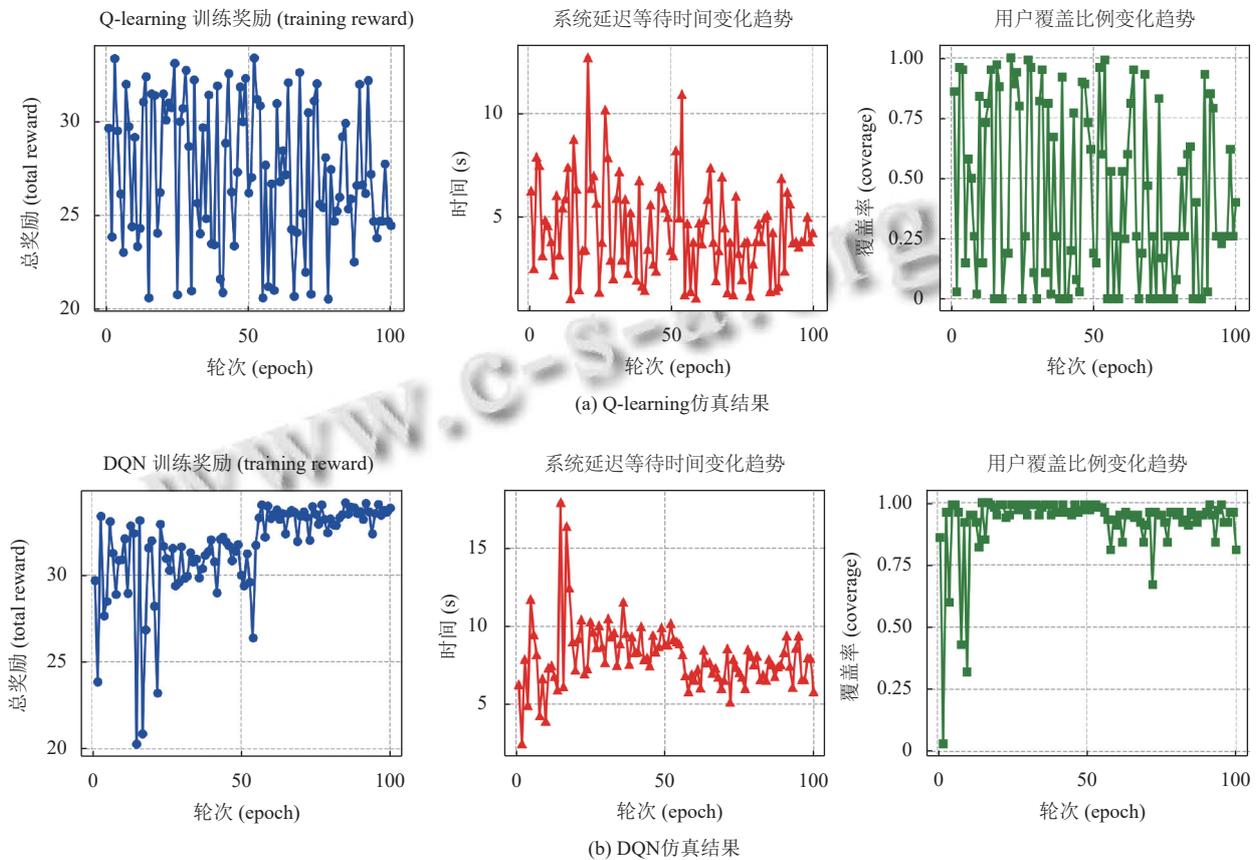


图5 强化学习仿真结果对比

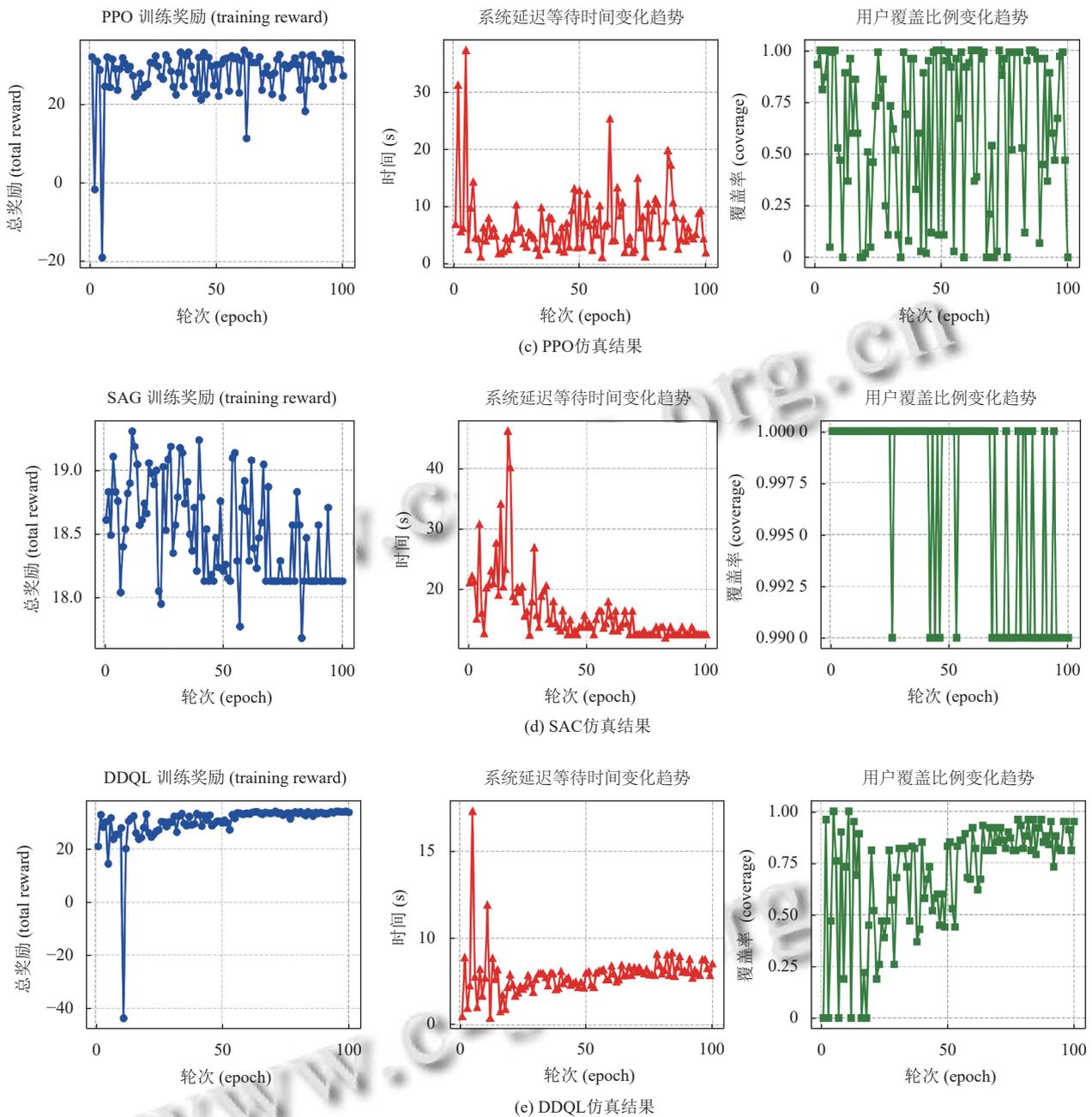


图5 强化学习仿真结果对比(续)

从仿真结果可以看出, DDQL 的奖励值收敛更快, 并且波动更小, 相较于其他算法, DDQL 在策略优化中更高效且稳定性更好, 并且在确保多数用户可以成功参与聚合的同时, 保证了 WFL 系统的整体性能。WFL 系统训练过程中的平均延迟等待时间对比和平均用户覆盖率对比如图 6 和图 7 所示。

从平均延迟等待时间来看, DDQL 仅用 5.7 s 的平均延迟等待时间, 就达到了 71% 的用户覆盖率, 完美实现了 WFL 系统下的延迟等待时间自适应调控, 其用

户覆盖比例也满足联邦学习结果可信性<sup>[25]</sup>的需求。同时 DDQL 相较于 SAC 节省了 65% 的延迟等待时间, 相较于 DQN 节省了 30% 的延迟等待时间, 可以完美保证 WFL 系统的高效运行。

同时, 本文对 CIFAR-10 和 MNIST 数据集的 IID 分区进行了多次测试。在 CIFAR-10 数据集上进行联邦聚合 FedAvg 的准确率变化和损失函数变化如图 8 和图 9 所示, 在 MNIST 数据集上进行联邦聚合 FedAvg 的准确率变化和损失函数变化如图 10 和图 11 所示,

仿真结果验证了基于 DDQL 进行自适应延迟等待时间调整算法的有效性。

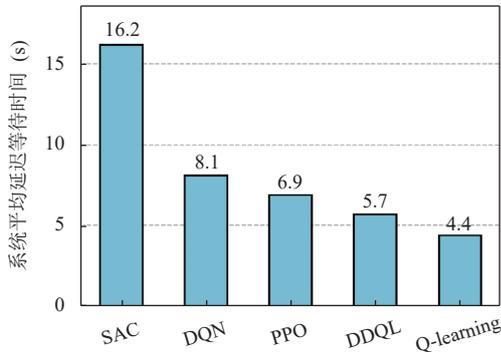


图 6 平均延迟等待时间对比图

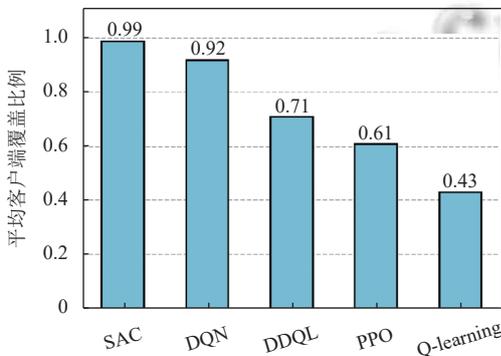


图 7 平均用户覆盖率对比图

在早期训练阶段, SAC 和 DQN 尝试通过设置更长的延迟等待时间来覆盖更多客户端, 因而在准确率上略有优势. 但这种策略导致系统性能消耗过高, 无法保持长期稳定性. 相比之下, DDQL 在初期更注重系统效率, 策略相对保守, 导致覆盖率和准确率略低于 SAC 和 DQN, 但是随着训练轮次增加, DDQL 对环境的认识逐渐加深, 通过自适应调控在保证稳定性能的同时逐步提升客户端覆盖率和模型精度, 最终在 CIFAR-10 和 MNIST 上都达到了与 SAC、DQN 相当甚至更高的准确率, 同时在资源消耗方面依旧保持着优势. 在数据集 CIFAR-10 上, DDQL 最终取得了 51.78% 的最高准确率, 相比 Q-learning 和 PPO 分别提升了 3.9% 和 3.1%; 在 MNIST 数据集上, 其准确率则达到 98.46%, 相比 Q-learning 和 PPO 分别提升了 0.4% 和 0.2%. 此外, DDQL 在损失值对比中收敛更快、更平稳, 且能保持约 71% 的平均用户覆盖率, 在保证联邦学习准确度的同时兼顾了训练的公平性. 这种自适应调整策略为 WFL 提供了更高的准确性与公平性, 确保了整体结果的可信性。

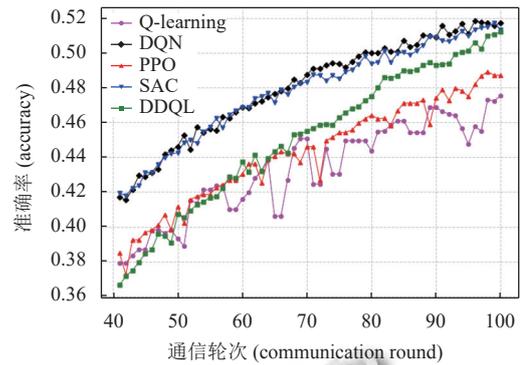


图 8 CIFAR-10 上的联邦学习准确率对比

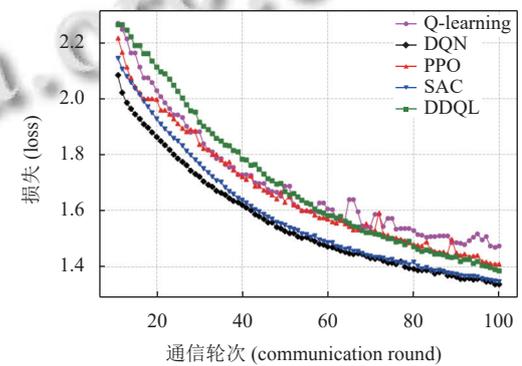


图 9 CIFAR-10 上的联邦学习损失对比

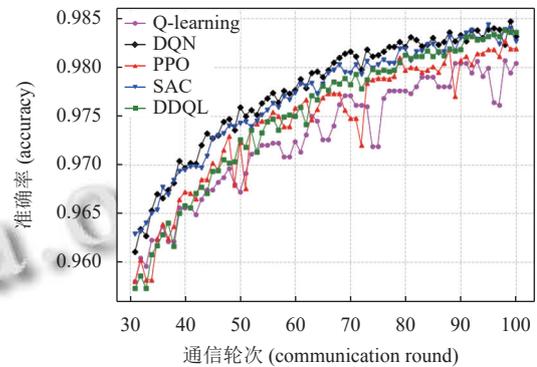


图 10 MNIST 上的联邦学习准确率对比

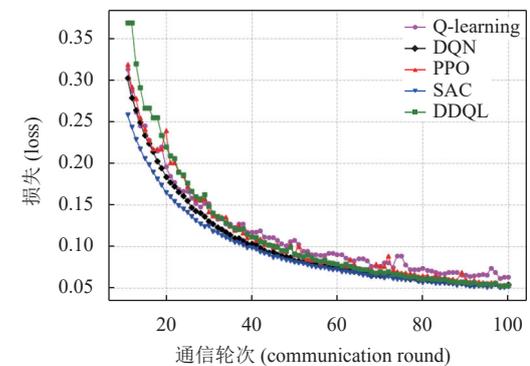


图 11 MNIST 上的联邦学习损失对比

## 4 结论与展望

本文提出一种新颖的自适应控制算法来解决 WFL 系统中由信道衰落造成的影响。该算法基于 DDQL 实现动态调整联邦学习中央聚合系统的延迟等待时间,综合考虑了联邦学习准确性、公平性和整体性能等多种因素,通过引入一种新颖的奖励函数来量化延迟等待时间变化对系统准确性、公平性和整体性能产生的影响,从而确定每一轮最优的动作调整,在保证 WFL 结果可信的同时不影响系统性能,能够有效缓解信道衰落造成的影响。此外,在 MNIST 和 CIFAR-10 数据集上的仿真结果表明,与现有基线强化学习算法相比,本文方法在保证 WFL 结果可信性和保证系统整体性能等方面均表现出显著优势。

展望未来,该算法可进一步结合更复杂的网络环境及多样化的数据场景,不断优化奖励函数设计并与其他强化学习策略融合,为 WFL 系统提供更具普适性和鲁棒性的自适应控制机制。同时,随着 5G、6G 等技术的发展,面向更高频段以及更具挑战性的信道条件的研究也将成为重点,以在更严苛的网络环境下持续确保 WFL 的准确性、公平性及整体性能。

### 参考文献

- 1 Gupta I, Singh AK, Lee CN, *et al.* Secure data storage and sharing techniques for data protection in cloud environments: A systematic review, analysis, and future directions. *IEEE Access*, 2022, 10: 71247–71277. [doi: [10.1109/ACCESS.2022.3188110](https://doi.org/10.1109/ACCESS.2022.3188110)]
- 2 McMahan HB, Moore E, Ramage D, *et al.* Communication-efficient learning of deep networks from decentralized data. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. Fort Lauderdale: PMLR, 2017. 1273–1282.
- 3 Xue JH, Qu Z, Xu J, *et al.* Bandwidth allocation for federated learning with wireless providers and cost constraints. *IEEE Transactions on Mobile Computing*, 2024, 23(6): 7470–7482. [doi: [10.1109/TMC.2023.3335258](https://doi.org/10.1109/TMC.2023.3335258)]
- 4 Ahmed KM, Imteaj A, Amini MH. Federated deep learning for heterogeneous edge computing. *Proceedings of the 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*. Pasadena: IEEE, 2021. 1146–1152. [doi: [10.1109/ICMLA52953.2021.00187](https://doi.org/10.1109/ICMLA52953.2021.00187)]
- 5 Amiri MM, Gündüz D. Federated learning over wireless fading channels. *IEEE Transactions on Wireless Communications*, 2020, 19(5): 3546–3557. [doi: [10.1109/TWC.2020.2974748](https://doi.org/10.1109/TWC.2020.2974748)]
- 6 王子璇, 吕娜, 王瀚璇, 等. 面向电力系统暂态稳定性的联邦学习拜占庭节点检测. *计算机系统应用*, 2024, 33(9): 235–244. [doi: [10.15888/j.cnki.csa.009578](https://doi.org/10.15888/j.cnki.csa.009578)]
- 7 Yang ZH, Chen MZ, Wong KK, *et al.* Federated learning for 6G: Applications, challenges, and opportunities. *Engineering*, 2022, 8: 33–41. [doi: [10.1016/j.eng.2021.12.002](https://doi.org/10.1016/j.eng.2021.12.002)]
- 8 Yang C, Cheng YQ, Yang XD, *et al.* A communication-efficient vertical federated learning via neuron selection. *Proceedings of the 2023 IEEE Smart World Congress (SWC)*. Portsmouth: IEEE, 2023. 1–8. [doi: [10.1109/SWC57546.2023.10449003](https://doi.org/10.1109/SWC57546.2023.10449003)]
- 9 Zheng P, Zhu Y, Hu YL, *et al.* Federated learning in heterogeneous networks with unreliable communication. *IEEE Transactions on Wireless Communications*, 2024, 23(4): 3823–3838. [doi: [10.1109/TWC.2023.3311824](https://doi.org/10.1109/TWC.2023.3311824)]
- 10 Wang YW, Gao Y, Liu XJ. Differential privacy hierarchical federated learning method based on privacy budget allocation. *Proceedings of the 9th International Conference on Computer and Communications (ICCC)*. Chengdu: IEEE, 2023. 2177–2181. [doi: [10.1109/ICCC59590.2023.10507299](https://doi.org/10.1109/ICCC59590.2023.10507299)]
- 11 Zeng SY, Mi B, Huang DR. Emergency vehicle identification for Internet of vehicles based on federated learning and homomorphic encryption. *Proceedings of the 12th IEEE Data Driven Control and Learning Systems Conference (DDCLS)*. Xiangtan: IEEE, 2023. 208–213. [doi: [10.1109/DDCLS58216.2023.10166254](https://doi.org/10.1109/DDCLS58216.2023.10166254)]
- 12 Beitollahi M, Lu N. Federated learning over wireless networks: Challenges and solutions. *IEEE Internet of Things Journal*, 2023, 10(16): 14749–14763. [doi: [10.1109/JIOT.2023.3285868](https://doi.org/10.1109/JIOT.2023.3285868)]
- 13 Jere S, Fan Q, Shang BD, *et al.* Federated learning in mobile edge computing: An edge-learning perspective for beyond 5G. *arXiv:2007.08030*, 2020.
- 14 Wang SY, Liu FF, Xia HL. Content-based vehicle selection and resource allocation for federated learning in IoV. *Proceedings of the 2021 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. Nanjing: IEEE, 2021. 1–7. [doi: [10.1109/WCNCW49093.2021.9419986](https://doi.org/10.1109/WCNCW49093.2021.9419986)]
- 15 Abbass W, Khan MA, Farooqi AH, *et al.* Optimizing spectrum utilization and security in SAS-enabled CBRS systems for enhanced 5G performance. *IEEE Access*, 2024, 12: 165992–166010. [doi: [10.1109/ACCESS.2024.3495972](https://doi.org/10.1109/ACCESS.2024.3495972)]
- 16 Chen ZJ, Liao GC, Ma Q, *et al.* Adaptive privacy budget

- allocation in federated learning: A multi-agent reinforcement learning approach. Proceedings of the 2024 IEEE International Conference on Communications. Denver: IEEE, 2024. 5166–5171. [doi: [10.1109/ICC51166.2024.10622685](https://doi.org/10.1109/ICC51166.2024.10622685)]
- 17 Liu ZM, Wang H, Li XB. FedUveQCS: Universal vector quantized compressive sensing for communication-efficient federated learning. IEEE Internet of Things Journal, 2024, 11(22): 37045–37056. [doi: [10.1109/JIOT.2024.3440959](https://doi.org/10.1109/JIOT.2024.3440959)]
- 18 田金箫. 提升联邦学习通信效率的梯度压缩算法. 计算机系统应用, 2022, 31(10): 199–205. [doi: [10.15888/j.cnki.csa.008748](https://doi.org/10.15888/j.cnki.csa.008748)]
- 19 Ma X, Wen CL, Wen T. An asynchronous and real-time update paradigm of federated learning for fault diagnosis. IEEE Transactions on Industrial Informatics, 2021, 17(12): 8531–8540. [doi: [10.1109/TII.2021.3063482](https://doi.org/10.1109/TII.2021.3063482)]
- 20 Bedda K, Fadlullah ZM, Fouda MM. Efficient wireless network slicing in 5G networks: An asynchronous federated learning approach. Proceedings of the 2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS). Bali: IEEE, 2022. 285–289. [doi: [10.1109/IoTaIS56727.2022.9976007](https://doi.org/10.1109/IoTaIS56727.2022.9976007)]
- 21 Tao B, Chen C, Chen HM. Communication efficient federated learning via channel-wise dynamic pruning. Proceedings of the 2023 IEEE Wireless Communications and Networking Conference (WCNC). Glasgow: IEEE, 2023. 1–6. [doi: [10.1109/WCNC55385.2023.10118879](https://doi.org/10.1109/WCNC55385.2023.10118879)]
- 22 李开菊, 许强, 王豪. 冗余数据去除的联邦学习高效通信方法. 通信学报, 2023, 44(5): 79–93. [doi: [10.11959/j.issn.1000-436x.2023072](https://doi.org/10.11959/j.issn.1000-436x.2023072)]
- 23 Khodak M, Balcan MF, Talwalkar A. Adaptive gradient-based meta-learning methods. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 5917–5928.
- 24 Chen TR, Zhang XR, You ML, *et al.* Federated learning enabled link scheduling in D2D wireless networks. IEEE Wireless Communications Letters, 2024, 13(1): 89–92. [doi: [10.1109/LWC.2023.3321500](https://doi.org/10.1109/LWC.2023.3321500)]
- 25 Shaban Z, Prasad R, Kumari P. Over the air federated learning in the presence of impulsive noise. Proceedings of the 2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW). Seoul: IEEE, 2024. 429–433. [doi: [10.1109/ICASSPW62465.2024.10625955](https://doi.org/10.1109/ICASSPW62465.2024.10625955)]
- 26 Bouaziz S, Benmeziane H, Imine Y, *et al.* FLASH-RL: Federated learning addressing system and static heterogeneity using reinforcement learning. Proceedings of the 41st IEEE International Conference on Computer Design (ICCD). Washington: IEEE, 2023. 444–447. [doi: [10.1109/ICCD58817.2023.00074](https://doi.org/10.1109/ICCD58817.2023.00074)]
- 27 Koursiompas N, Magoula L, Petropouleas N, *et al.* A safe deep reinforcement learning approach for energy efficient federated learning in wireless communication networks. IEEE Transactions on Green Communications and Networking, 2024, 8(4): 1862–1874. [doi: [10.1109/TGCN.2024.3372695](https://doi.org/10.1109/TGCN.2024.3372695)]
- 28 Zhang SQ, Lin JY, Zhang Q. A multi-agent reinforcement learning approach for efficient client selection in federated learning. Proceedings of the 36th AAAI Conference on Artificial Intelligence. AAAI Press, 2022. 9091–9099.
- 29 Mao W, Lu XJ, Jiang YH, *et al.* Joint client selection and bandwidth allocation of wireless federated learning by deep reinforcement learning. IEEE Transactions on Services Computing, 2024, 17(1): 336–348. [doi: [10.1109/TSC.2024.3350050](https://doi.org/10.1109/TSC.2024.3350050)]
- 30 Li PC, Xiao YY, Yan JH, *et al.* Reinforcement learning for adaptive resource scheduling in complex system environments. Proceedings of the 5th International Symposium on Computer Engineering and Intelligent Communications (ISCEIC). Wuhan: IEEE, 2024. 92–98. [doi: [10.1109/ISCEIC63613.2024.10810157](https://doi.org/10.1109/ISCEIC63613.2024.10810157)]
- 31 Pi Y, Zhang W, Zhang Y, *et al.* Applications of multi-agent deep reinforcement learning communication in network management: A survey. arXiv:2407.17030, 2024.
- 32 Zhang YQ, Xia H, Xu S, *et al.* AdaptFL: Adaptive federated learning framework for heterogeneous devices. Future Generation Computer Systems, 2025, 165: 107610. [doi: [10.1016/j.future.2024.107610](https://doi.org/10.1016/j.future.2024.107610)]
- 33 Wu SL, Shen S, Yeoh PL, *et al.* Deep reinforcement learning-empowered federated learning for wireless clients with energy and bandwidth constraints. Proceedings of the 2024 IEEE Annual Congress on Artificial Intelligence of Things (AIoT). Melbourne: IEEE, 2024. 1–6. [doi: [10.1109/AIoT63253.2024.00011](https://doi.org/10.1109/AIoT63253.2024.00011)]
- 34 Luo ZN, Hu FZ. Simulation models for independent Rayleigh fading channels. Proceedings of the 2010 Global Mobile Congress. Shanghai: IEEE, 2010. 1–5. [doi: [10.1109/GMC.2010.5634573](https://doi.org/10.1109/GMC.2010.5634573)]
- 35 Chen MZ, Yang ZH, Saad W, *et al.* A joint learning and communications framework for federated learning over wireless networks. IEEE Transactions on Wireless Communications, 2021, 20(1): 269–283. [doi: [10.1109/TWC.2020.3024629](https://doi.org/10.1109/TWC.2020.3024629)]

- 36 Qi P, Chiaro D, Guzzo A, *et al.* Model aggregation techniques in federated learning: A comprehensive survey. *Future Generation Computer Systems*, 2024, 150: 272–293. [doi: [10.1016/j.future.2023.09.008](https://doi.org/10.1016/j.future.2023.09.008)]
- 37 van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning. *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. Phoenix: AAAI Press, 2015. 2094–2100.
- 38 Soni H, Vyas R, Hiran KK. Self-autonomous car simulation using deep Q-learning algorithm. *Proceedings of the 2022 International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT)*. Pune: IEEE, 2022. 1–4. [doi: [10.1109/TQCEBT54229.2022.10041614](https://doi.org/10.1109/TQCEBT54229.2022.10041614)]
- 39 Li RQ, Su S, Gu YK, *et al.* Energy-efficient train control method based on batch constrained deep Q-learning. *Proceedings of the 26th IEEE International Conference on Intelligent Transportation Systems (ITSC)*. Bilbao: IEEE, 2023. 2920–2925. [doi: [10.1109/ITSC57777.2023.10422583](https://doi.org/10.1109/ITSC57777.2023.10422583)]
- 40 Schulman J, Wolski F, Dhariwal P, *et al.* Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.
- 41 Haarnoja T, Zhou A, Abbeel P, *et al.* Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *Proceedings of the 35th International Conference on Machine Learning*. Stockholm: PMLR, 2018. 1856–1865.
- 42 LeCun Y, Cortes C, Burges CJC. The MNIST database of handwritten digits. <http://yann.lecun.com/exdb/mnist/>. [2024-12-20].
- 43 Krizhevsky A. Learning multiple layers of features from tiny images [Technical Report]. Toronto: University of Toronto, 2009. <https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf>.
- 44 Zeng D, Liang SQ, Hu XJ, *et al.* FedLab: A flexible federated learning framework. *The Journal of Machine Learning Research*, 2023, 24(1): 4614–4620.
- 45 Wu WT, He LG, Lin WW, *et al.* FedProf: Optimizing federated learning with dynamic data profiling. *arXiv:2102.01733*, 2021.

(校对责编: 张重毅)