

基于 FL 的无蜂窝网络用户调度与功率分配策略

王华华, 黄烨霞, 李玲

(重庆邮电大学通信与信息工程学院, 重庆 400065)

摘要: 为解决无蜂窝网络系统中用户链路质量差异和通信、计算资源占用不平衡导致的联邦学习 (FL) 训练性能受限问题, 设计了一个联合用户调度和功率分配的优化方案。首先, 提出了一种低复杂度的基于资源优先的二次抽样用户调度 (RPSS-US) 算法, 根据用户计算资源的可用性和链路质量选择用户, 优先调度对系统容量和全局模型更新贡献较大的用户参与 FL 任务, 提高整体训练性能。随后, 提出了一种基于二分法的功率分配 (BM-PA) 算法, 通过优化功率分配改善用户链路质量差异, 以提高数据传输速率, 减少 FL 任务的总体时延。通过交替迭代优化这两个子问题, 实现系统性能的联合优化。仿真结果表明, 相较于其他对比算法, 所提出的算法下行吞吐量提升了 47.19%, 上行吞吐量提升了 22.60%, FL 任务时间消耗减少了 57.33%, 并在达到相同模型精度时的时间开销最小。

关键词: 无蜂窝网络; 联邦学习; 用户调度; 功率分配; 联合优化

中图分类号: TN929.5

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2024159

User scheduling and power allocation strategy for cell-free networks based on federated learning

WANG Huahua, HUANG Yexia, LI Ling

School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Abstract: In order to address the issue of limited training performance in federated learning (FL) due to user link quality disparities and imbalanced communication, and computing resource utilization in cell-free network systems, a joint optimization problem for user scheduling and power allocation was designed. Firstly, a low-complexity resource priority based secondary sampling user scheduling (RPSS-US) algorithm was proposed. Users were selected based on the availability of their computing resources and link quality, with priority given to those contributing more to system capacity and global model updates, thus improving overall training performance. Then, a power allocation algorithm based on the binary method (BM-PA) was proposed to optimize power allocation, improve user link quality differences, enhance data transmission rates, and reduce overall FL task delay. By iteratively optimizing these two sub-problems alternately, joint optimization of system performance was achieved. Simulation results demonstrate that compared to other comparison algorithms, the proposed algorithm achieves a 47.19% increase in downlink throughput, a 22.60% increase in uplink throughput, and a 57.33% reduction in FL task time consumption, while minimizing time overhead for achieving the same model accuracy

Keywords: cell-free network, federated learning, user scheduling, power allocation, joint optimization

收稿日期: 2024-04-10; 修回日期: 2024-06-13

通信作者: 黄烨霞, huangyexia2022@163.com

基金项目: 重庆市自然科学基金创新发展联合基金(中国星网)资助项目(No.CSTB2023NSCQ-LZX0114)

Foundation Item: Chongqing Natural Science Foundation Innovation and Development Joint Fund (China Star Network) (No.CSTB2023NSCQ-LZX0114)

0 引言

近年来,随着移动设备、可穿戴设备和自动驾驶汽车的普及,无线网络系统产生了大量数据,推动了机器学习(ML, machine learning)技术在该领域的迅速发展。然而,传统的ML框架依赖于集中式结构,用户设备(UE, user equipment)收集的原始数据需要在云端集中存储和处理,这导致数据隐私泄露风险、传输时延和能耗等问题。为了解决这些问题,联邦学习^[1](FL, federated learning)作为一种新兴的分布式机器学习框架备受关注,FL允许用户在本地设备上对模型进行训练,只将模型参数上传至中央服务器进行聚合,从而有效保护用户隐私。与此同时,避免了传输原始数据(数据量远大于模型参数)的成本,从而降低了系统开销和能量消耗。

尽管FL具有诸多优势,但由于FL需要在UE和服务器之间频繁交换模型参数,一旦服务器聚合了错误的参数,在后续迭代中错误会不断累积,严重影响模型性能。因此,在实际无线网络环境中,有限的无线资源和无线链路的不可靠性成为FL性能提升的瓶颈。蜂窝大规模多输入多输出(CF mMIMO, cell-free massive multiple-input and multiple-output)系统^[2-4]成为解决上述问题的方案之一,通过部署大量网络接入点(AP, access point),利用信道硬化和有利传输的特性,能在有限区域内提供高质量的通信链路,保障了FL迭代过程中的通信稳定性。CF mMIMO结合联邦学习技术,可以在不增加数据传输负担的情况下,利用分布式数据进行高效的模型训练,进一步提升网络性能和服务质量^[5]。然而,FL中存在一个限制,即只有当所有UE成功地将参数传输到中央服务器时,中央服务器才能有效地聚合全局模型,特别是在同步FL中^[6]。因此,计算资源有限或具有不良链路的UE(也称为“落伍者”)会导致中央服务器长时间等待,严重减缓整个FL任务的进程。在CF mMIMO系统中,大量的AP和UE增加了“落伍者”出现的概率。因此,如何在CF mMIMO系统中选择最佳的UE参与FL过程,避免“落伍者”效应,并通过资源分配优化方案提升网络性能,成为一个值得深入研究的问题。

为了缓解“落伍者”效应,一种常见的方法是在UE的子集进行FL本地模型参数更新,从而降

低“落伍者”参与FL任务的概率。许多专家学者^[7-9]对UE调度方案展开了研究,例如,文献[7]提出一种在线UE选择方案,旨在每次迭代中最小化训练时间并确保收敛性。文献[8]根据信道条件优化每个AP所服务的UE数目。此外,文献[9]研究了基于回程链路功耗和频谱效率的接入点选择方案,特别适用于大量接入点的场景,显著提高了总能量效率。然而,当UE间的数据异质性较高时,过度追求每轮迭代中UE的数量可能会损害FL模型的精度。文献[10]则利用反向传播(BP, back propagation)神经网络将AP与UE之间的大尺度衰落作为输入,提出了一种基于BP神经网络的AP动态选择算法,并通过改进遗传算法用于AP选择,从而提高了系统的频谱和能量效率。然而,这种方法可能会牺牲系统的复杂度,并且仍然存在UE成为“落伍者”的潜在风险。因此,仅依赖UE调度不足以完全有效避免“落伍者”效应,还需综合考虑FL在无线网络中的资源调度与分配问题。

文献[11-12]深入探讨了联合带宽分配和设备调度问题,以最大化相对于时间的收敛速率。文献[13]将无线网络中FL过程中的功率、无线带宽资源分配和用户选择问题建模为一个优化问题,通过最小化FL损失函数来捕捉FL算法的性能。而文献[14]则以降低基站和用户端的能耗为目标优化了功率和计算资源的分配。然而,这些算法在蜂窝网络系统中的实际有效性尚待进一步验证。于是,文献[15]提出了一种基于加速投影梯度(APG, accelerated projected gradient)方法的功率控制算法,用于解决CF mMIMO系统中的能量效率最大化问题。该算法考虑了AP的功率限制和用户的服务质量(QoS, quality of service)要求,并能够有效地找到最优的功率分配方案。尽管上述方法在一定程度上提升了系统性能,但是由于模型复杂度较高,在实际应用中可能受到限制。

近年来,学者们开始探索在蜂窝网络中实现FL的有效网络架构^[5,16-18]。其中,文献[5]通过构建优化问题,联合优化了模型精度、发射功率、数据速率和用户中央服务器(CPU, central processing unit)处理频率,并引入了凹下界以处理非凸约束条件并迭代求解。然而,该研究未充分考虑用户调度策略对系统效率和性能提升的贡献。文献[16]针对此问题提出了基于信道排序的较差用户优先接入策略,

通过按照信道状态质量顺序选择合适的接入点,实现了系统总能耗的最小化。文献[17]和文献[18]进一步研究了用户选择、发射功率分配的联合优化问题,通过选择连接状况良好(即,信道增益高、导频污染和干扰弱)的用户,以减少每个FL通信轮次的执行时间。然而,这些研究主要关注具有有利链路用户的选择,忽视了用户持有的计算资源对系统学习性能的影响。

综上所述,目前在无蜂窝网络系统中执行FL任务的调度与资源分配策略研究仍存在不足。现有研究过度关注通信资源(如信噪比、发射功率、带宽等)在用户选择时的比重,而对与计算相关的资源(如数据集大小、数据集的分布)缺乏考虑。虽然连接状况良好的用户可以更快地完成数据传输和计算任务,但这也意味着一些连接状况较差的用户被排除在外。这些用户可能拥有更丰富的数据样本或更符合模型目标的数据分布,而他们的参与有助于提升模型精度。因此,仅仅以用户通信资源的丰富程度作为调度时的考量准则,可能需要增加更多FL通信轮次来达到收敛,反而导致FL过程收敛速度和效率的下降。这样的方案虽然能够提升通信性能,但却以牺牲学习性能为代价。

为了在系统通信性能与学习性能之间实现有效权衡,本文深入研究了在无蜂窝网络系统中执行FL任务时的资源调度与分配问题,联合设计了一个涉及UE选择、上下行功率分配的优化问题,并提出了一种基于资源优先的二次抽样用户调度(RPSS-US, resource priority based secondary sampling user scheduling)算法和基于二分法的功率分配(BM-PA, power allocation based on binary method)算法分别对用户调度和功率分配2个子问题进行求解,旨在通过联合优化用户调度和功率分配,最小化FL过程的执行时间,并提升系统的通信与学习性能。

本文的主要贡献如下。

1) 在FL框架下,针对无蜂窝网络系统,建立了时间消耗模型,并提出了一种综合考虑用户信道质量和数据重要性的RPSS-US算法,通过评估每个UE持有的通信资源与计算资源对系统容量和全局模型更新贡献的大小,动态选择参与每轮FL迭代的UE。该方法不仅能够削弱“落伍者”效应,还有效平衡了系统的通信性能和学习性能。

2) 基于上述调度策略,本文进一步提出了BM-PA算法,该算法在保持较低复杂度的同时,实现了对已关联用户上行和下行链路的功率分配优化,从而提升系统容量并减少FL的总时耗。

3) 通过与其他对比方案进行比较,仿真结果表明,本文所提算法在优化系统吞吐量方面效果显著,同时显著降低系统的总时间消耗,为无蜂窝网络系统中FL任务的执行提供了有效的优化策略。

1 系统模型

本文考虑一个采用时分多址(TDMA, time division multiple access)模式的无蜂窝网络系统,网络中大小为 M 的AP集 $\mathcal{M} \triangleq \{1, 2, \dots, M\}$ 为大小为 K 的UE集 $\mathcal{K} \triangleq \{1, 2, \dots, K\}$ 提供服务。AP和UE均匀地分布在一个 $D_m \times D_m$ 的区域内,且AP和UE均为单天线。用户通过无线链路与AP进行连接,AP则通过高容量的回程链路连接到CPU,因此本文不考虑CPU与所有AP之间的传输时延。在一个FL训练任务中,CPU和UE分别作为中央服务器和客户端,AP作为中继在CPU和UE之间传递模型的更新参数。系统主要参数如表1所示。

表1 系统主要参数

参数	含义
\mathcal{M}, M	AP集及其大小
\mathcal{K}, K	UE集及其大小
τ_c, τ_p	相干时间长度和导频序列长度
p_p, p_d, p_u	导频、下行和上行归一化噪声功率
S_d, S_u	下发和上传数据量的大小
\mathcal{D}_k, D_k	第 k 个UE持有的训练数据集及其大小
a_k	指示第 k 个UE是否被选择的二元变量
η_{mk}	AP m 到UE k 的下行功率控制因子
ς_k	UE k 的上行功率控制因子

首先定义一个二元变量 a_k 指示UE k 是否被关联,定义 $\hat{\mathcal{U}}$ 表示最终参与的FL任务的用户集,当 $k \in \hat{\mathcal{U}}$ 时 $a_k = 1$,否则 $a_k = 0$ 。

FL是一个迭代过程,在每一轮通信中,中央服务器向UE广播全局模型,UE接收并利用本地数据计算本地模型更新参数,然后将更新参数发回中央服务器进行聚合。FL的核心思想是在本地设备上模型训练,只将模型参数上传至中央服务器进行聚合,从而避免了原始数据的共享。本文假设在无蜂窝网络系统中,被关联的UE参与一个采

用同步聚合模式的标准 FL 框架^[19], 即只有在接收到所有被关联用户子集的所有用户上传的模型参数后才开始聚合。每一个通信回合中的 FL 任务包含以下 4 个基本步骤。

步骤 1 模型下载 (S1)。CPU 执行用户调度算法得到本轮的调度策略, 并将该策略和全局更新模型参数以广播的形式下发给所有 AP。由 AP 将全局更新模型参数传达给需要关联的 UE。

步骤 2 模型更新 (S2)。关联 UE 接收到全局模型参数, 并利用本地数据集计算并更新本地模型参数。

步骤 3 模型上传 (S3)。所有被关联的 UE 向 AP 上传其本地训练模型更新参数并报告其数据集大小、存储容量、CPU 处理频率等状态信息, 以便为服务器执行用户选择提供信息支撑。最后由 AP 将从 UE 获取的所有数据上传至服务器。

步骤 4 模型聚合 (S4)。CPU 在接收到所有 AP 发来的数据后, 采用 FedAvg 算法^[20]更新全局模型参数。

不断迭代执行上述步骤, 直到收敛时终止整个 FL 任务。基于 FL 的 CF mMIMO 无线网络模型如图 1 所示。

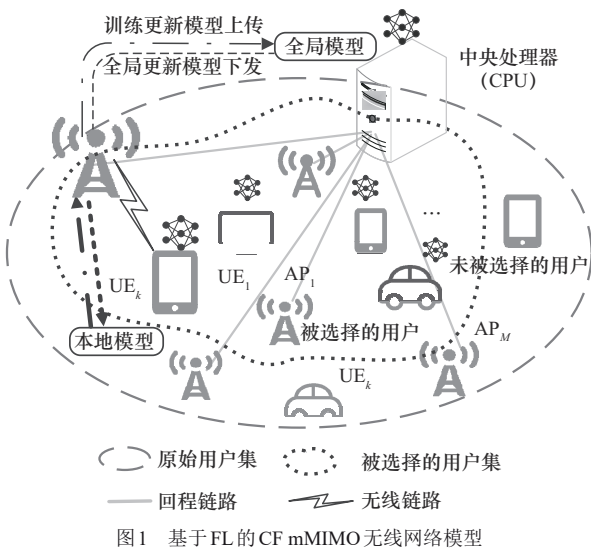


图 1 基于 FL 的 CF mMIMO 无线网络模型

为了简化模型, 做出如下假设。

假设 1: 每轮 FL 的训练都在一个大尺度相干时间内完成。

假设 2: 所有关联 UE 以同步的方式执行 S2 和 S3。聚合工作在接收到所有关联 UE 的本地更新模型后才进行。

S1 分为 2 个阶段: 下行链路信道估计和下行数据传输。类似地, S3 也分为 2 个阶段: 上行链路信道估计和上行数据传输。所有 4 个阶段都在一个相干时间块内完成。利用 TDMA 模式信道互易的特性本文 S3 的信道估计结果采用 S1 的信道估计结果。在整个的通信过程中, FL 任务时间阶段划分情况如图 2 所示。

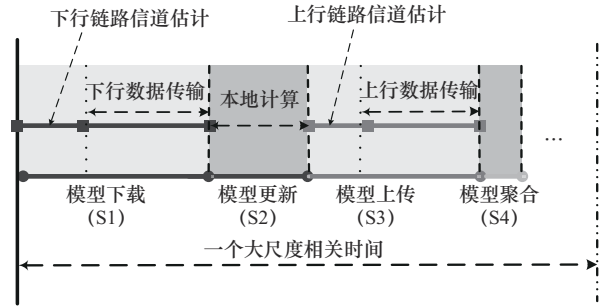


图 2 FL 任务时间阶段划分情况

系统上行链路信道估计、下行数据传输和上行数据传输模型将在下面进行描述。

1.1 上行链路信道估计模型

在上行链路信道估计阶段, UE 向 AP 发送长度为 $\tau_p \ll \tau_c$ 的导频序列, τ_c 表示每个相干块能够传送的样本数量。每个 AP 利用这些导频序列进行信道估计。

假设 UE_k 发送的导频序列为 $\sqrt{\tau_p} \boldsymbol{\varphi}_k$, 其中 $\|\boldsymbol{\varphi}_k\|^2 = 1, \forall k$, 则在第 m 个 AP 处接收的导频信号向量为

$$\mathbf{y}_{p,m} = \sqrt{\tau_p} p_p \sum_{k=1}^K \mathbf{g}_{mk} \boldsymbol{\varphi}_k + \mathbf{n}_{p,m} \quad (1)$$

其中, p_p 是导频序列的归一化发射功率; $\boldsymbol{\varphi}_k$ 是第 k 个 UE 使用的正交导频序列; $\mathbf{n}_{p,m}$ 是第 m 个 AP 处的加性噪声; $\mathbf{g}_{mk} = \sqrt{\beta_{mk}} h_{mk}$ 表示从 UE_k 到 AP_m 的信道, β_{mk} 和 h_{mk} 分别代表信道大尺度和小尺度衰落系数。AP_m 基于接收的导频信号 $\mathbf{y}_{p,m}$ 采用最小均方误差估计 (MMSE, minimum mean squared error) 方法对 \mathbf{g}_{mk} 进行估计得到估计值 $\hat{\mathbf{g}}_{mk} \sim \text{CN}(0, \sigma_{mk}^2)$, 其中, σ_{mk}^2 的具体表达式为

$$\sigma_{mk}^2 = \frac{\tau_p p_p (\beta_{mk})^2}{\tau_p p_p \sum_{i \in \mathcal{K}} \beta_{mi} |\boldsymbol{\varphi}_k^H \boldsymbol{\varphi}_i|^2 + 1} \quad (2)$$

其中, $\mathcal{K} \triangleq \{1, 2, \dots, K\}$ 代表 UE 集合。

1.2 下行数据传输模型

获得信道估计信息后, AP使用共轭波束成形技术向用户发送数据信号。令 $s_{d,k}$ 表示CPU需要下发的全局模型参数编码符号。则在 AP_m 处的发送信号可表示为

$$x_{d,m} = \sqrt{p_d} \sum_{k \in \mathcal{K}} \sqrt{\eta_{mk}} (\hat{g}_{mk})^* s_{d,k} \quad (3)$$

其中, p_d 是每个AP处的最大归一化发射功率, $0 \leq \eta_{mk} \leq 1$ 是下行功率控制因子。 AP_m 处的发射功率需要满足平均归一化功率约束 $\mathbb{E} \{ |x_{d,m}|^2 \} \leq p_d$, 即 $\sum_{k \in \mathcal{K}} \sigma_{mk}^2 \eta_{mk} \leq 1, \forall m$ 。则 UE_k 处的接收信号为

$$\begin{aligned} y_{d,k} &= \sum_{m \in \mathcal{M}} g_{mk} x_{d,m} + n_k = \\ &\sqrt{p_d} \sum_{m \in \mathcal{M}} \sqrt{\eta_{mk}} g_{mk} (g_{mk})^* s_{d,k} + \\ &\sqrt{p_d} \sum_{m \in \mathcal{M}} \sum_{i \in \mathcal{K}} \sqrt{\eta_{mi}} g_{mk} (\hat{g}_{mi})^* s_{d,i} + n_k \end{aligned} \quad (4)$$

其中, n_k 是 UE_k 的加性噪声, $\mathcal{M} \triangleq \{1, 2, \dots, M\}$ 代表AP集合。则 UE_k 的下行可达速率为

$$\begin{aligned} R_{d,k}(a_k, \boldsymbol{\eta}) &= \frac{\tau_c - \tau_p}{\tau_c} B^D \text{lb} \\ &\left(1 + \frac{p_d \left(\sum_{k \in \mathcal{K}} \sum_{m \in \mathcal{M}} a_k \sqrt{\eta_{mk}} \sigma_{mk}^2 \right)^2}{p_d \sum_{k' \in \mathcal{K}} a_{k'} A_{kk'}^2 |\boldsymbol{\varphi}_k^H \boldsymbol{\varphi}_{k'}|^2 + p_d \sum_{k \in \mathcal{K}} a_k B_k + 1} \right) \end{aligned} \quad (5)$$

其中, B^D 为下行传输带宽, 二元变量 a_k 用于指示 UE_k 是否被关联, k' 代表UE集合中除编号为 k 之外其他UE的编号, $A_{kk'} = \sqrt{\eta_{mk'}} \sigma_{mk'}^2 \left(\frac{\beta_{mk'}}{\beta_{mk'}} \right)$, $B_k =$

$$\sum_{m=1}^M \eta_{mk} \sigma_{mk}^2 \beta_{mk}^\circ$$

1.3 上行数据传输模型

在上行数据传输阶段, 令 $s_{u,k}$ 表示 UE_k 的本地更新模型参数编码符号, 则 UE_k 处需要上传的基带信号表示为

$$x_{u,k} = \sqrt{p_u \zeta_k} s_{u,k} \quad (6)$$

其中, p_u 是 UE_k 处的最大归一化发射功率, ζ_k 是上行功率控制因子。同样地, UE的发送信号也需要满足平均发射功率约束: $\mathbb{E} \{ |x_{u,k}|^2 \} \leq p_u$, 即 $0 \leq \zeta_k \leq 1, \forall k$ 。则 AP_m 处的接收信号可表示为

$$\begin{aligned} y_{u,m} &= \sum_{k \in \mathcal{K}} g_{mk} x_{u,k} + n_{u,m} = \\ &\sqrt{p_u} \sum_{k \in \mathcal{K}} \sqrt{\zeta_k} g_{mk} s_{u,k} + n_{u,m} \end{aligned} \quad (7)$$

其中, $n_{u,m}$ 是 AP_m 处的加性噪声。为了检测从第 k 个用户发送的符号 $s_{u,k}$, AP将接收信号 $y_{u,m}$ 与其信道估计 \hat{g}_{mk} (在上行链路信道估计时获得)的共轭相乘, 然后通过回程链路发送到CPU, 此时CPU处的接收信号可表示为

$$\begin{aligned} y_{u,k} &= \sqrt{p_u} \sum_{m \in \mathcal{M}} \sqrt{\zeta_k} g_{mk} (\hat{g}_{mk})^* s_{u,k} + \\ &\sqrt{p_u} \sum_{m \in \mathcal{M}} \sum_{k' \in \mathcal{K}} \sqrt{\zeta_{k'}} g_{mk} (\hat{g}_{mk'})^* s_{u,k'} + \\ &\sum_{m \in \mathcal{M}} (\hat{g}_{mk})^* n_{u,m} \end{aligned} \quad (8)$$

因此, UE_k 可实现的上行传输速率为

$$\begin{aligned} R_{u,k}(a_k, \boldsymbol{\zeta}) &= \frac{\tau_c - \tau_p}{\tau_c} B^U \cdot \\ &\text{lb} \left(1 + \frac{p_u a_k \zeta_k \left(\sum_{m \in \mathcal{M}} \sigma_{mk}^2 \right)^2}{p_u \sum_{k' \in \mathcal{K}} a_{k'} U_{kk'}^2 |\boldsymbol{\varphi}_k^H \boldsymbol{\varphi}_{k'}|^2 + p_u \sum_{k \in \mathcal{K}} a_k V_k + W_k} \right) \end{aligned} \quad (9)$$

其中, B^U 为上行传输带宽, $U_{kk'} = \sum_{m \in \mathcal{M}} \zeta_{k'} \sigma_{mk}^2 \left(\frac{\beta_{mk'}}{\beta_{mk}} \right)$,

$$V_k = \sum_{m=1}^M \zeta_k \sigma_{mk}^2 \beta_{mk}, \quad W_k = \sum_{m \in \mathcal{M}} \sigma_{mk}^2$$

1.4 用户调度和功率分配联合优化模型

在基于联邦学习框架的无蜂窝网络系统中, 本文以最小化系统总时延为目标, 设计了一个联合用户调度和功率分配的优化问题, 旨在寻找一个合理的UE子集与功率分配方案, 以实现网络质量和FL性能的双重提升。

本节对系统时间消耗模型进行假设分析。在考虑用户调度策略的情况下, 假设全局训练模型和本场更新模型数据大小分别为 S_d 和 S_u (单位为比特), 则下行和上行的传输时延可建模为

$$t_{d,k}(a_k, \boldsymbol{\eta}) = \frac{a_k S_d}{R_{d,k}(a_k, \boldsymbol{\eta})} \quad (10)$$

$$t_{u,k}(a_k, \boldsymbol{\zeta}) = \frac{a_k S_u}{R_{u,k}(a_k, \boldsymbol{\zeta})} \quad (11)$$

UE_k 处的计算更新时延为

$$t_{c,k}(a_k) = \frac{a_k L D_k c}{f} \quad (12)$$

其中, L 为UE本地模型参数更新需要的迭代总次

数, c (周期/样本) 表示 UE_k 处理一个数据样本需要的周期数, f 是用户 k 的 CPU 处理频率 (周期/秒), D_k 代表 UE_k 的数据集大小。

在每个通信回合中, 根据 AP 所处的传输模式可将其分为上行传输阶段和下行传输阶段。如图 3 所示, 在下行传输阶段, 所耗费的时间取决于最后完成步骤 1 和步骤 2 的 UE 所需的时间, 而在上行传输阶段, 时延则取决于最后完成步骤 3 的 UE 所需的时间。

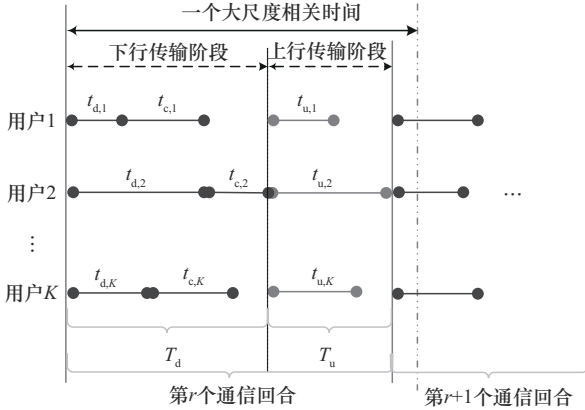


图 3 系统时间消耗模型

因此, 一个通信回合的时间消耗 T_o 可由式(13)~式(15)求得。

$$T_d(\mathbf{a}, \boldsymbol{\eta}) = \max_{k \in \mathcal{K}} \left\{ \frac{a_k S_d}{R_{d,k}(\boldsymbol{\eta})} + \frac{a_k L D_k c}{f} \right\} \quad (13)$$

$$T_u(\mathbf{a}, \boldsymbol{\varsigma}) = \max_{k \in \mathcal{K}} \frac{a_k S_u}{R_{u,k}(\boldsymbol{\varsigma})} \quad (14)$$

$$T_o(\mathbf{a}, \boldsymbol{\eta}, \boldsymbol{\varsigma}) = T_d(\mathbf{a}, \boldsymbol{\eta}) + T_u(\mathbf{a}, \boldsymbol{\varsigma}) \quad (15)$$

根据文献[21], 可得整个 FL 过程需要的通信轮数为

$$Q(\mathbf{a}) = \frac{q}{\sum_{k \in \mathcal{K}} a_k} \quad (16)$$

其中, q 是一个取决于具体 FL 任务的已知常数。则整个 FL 任务的总执行时间为

$$T_e(\mathbf{a}, \boldsymbol{\eta}, \boldsymbol{\varsigma}) = Q(\mathbf{a}) T_o(\mathbf{a}, \boldsymbol{\eta}, \boldsymbol{\varsigma}) \quad (17)$$

式(13)~式(17)反映了 UE 选择对总执行时间的影响。根据式(13)~式(15), 通过仅允许具有良好链路条件的用户 (即具有较强信道增益、较低导频污染和较低干扰的用户) 参与 FL 过程, 可以缩短一个 FL 通信回合的执行时间。然而, 由式(16)可知, 较少的 UE 参与训练也意味着所需的通信回合数增加。因此, 最小化式(17)中的总执行时间就涉及寻找一个最优的候选 UE 集合, 以平衡这 2 种相互矛

盾的影响。

整个联合优化问题可以建模为

$$\begin{aligned} & \min_{\mathbf{a}, \boldsymbol{\eta}, \boldsymbol{\varsigma}} T_e(\mathbf{a}, \boldsymbol{\eta}, \boldsymbol{\varsigma}) \\ \text{s.t. } & C_1: \frac{a_k S_d}{R_{d,k}(\mathbf{a}, \boldsymbol{\eta})} + \frac{a_k L D_k c}{f} + \frac{a_k S_u}{R_{u,k}(\mathbf{a}, \boldsymbol{\varsigma})} < T_{\text{round}} \\ & C_2: a_k = \begin{cases} 1, k \in \hat{\mathcal{U}} \\ 0, k \notin \hat{\mathcal{U}} \end{cases} \\ & C_3: \sum_{k \in \mathcal{K}} a_k \geq N_{\text{th}} \\ & C_4: 0 \leq \zeta_k \leq 1 \\ & C_5: \sum_{k \in \mathcal{K}} \sigma_{mk}^2 \eta_{mk} \leq 1 \\ & C_6: 0 \leq \eta_{mk} \end{aligned} \quad (18)$$

其中, T_{round} 代表一个 FL 通信回合允许的最大执行时间, 超过最大执行时间的轮次将被视为无效的通信回合; N_{th} 是为了达到模型精度所需最少参与用户数目, 通过保持参与 FL 过程的 UE 的数量大于某个值 N_{th} 来保证 FL 的质量。问题式(18)具有非凸随机性、混合整数混合时间尺度、二元约束和变量间紧耦合等特点。为了解决这一优化问题, 将其分解为用户调度子问题和资源分配子问题, 并采用交替优化的方式来解决。在问题式(18)中, UE 选择变量在 \mathbf{a} 执行 FL 过程之前被优化, 在一个相干时间内保持不变, 而功率控制因子 $\boldsymbol{\eta}$ 和 $\boldsymbol{\varsigma}$ 则在每个 FL 通信回合中进行优化, 经过多轮迭代优化最终得到最优的结果 $\boldsymbol{\eta}^*$ 和 $\boldsymbol{\varsigma}^*$ 。具体细节将在后续章节中详细阐述。

2 算法设计

2.1 基于资源优先的用户调度算法

本节详细阐述无蜂窝网络系统中支持联邦学习的用户调度策略。该策略以最小化完成一个 FL 任务的总时间为目标, 提出了一种低复杂度的 RPSS-US 算法。该算法通过两次抽样筛选出最终参与 FL 任务的用户, 第一次抽样以用户计算资源的丰富程度为准则, 第二次抽样则以用户持有的通信资源的丰富程度为考量指标。通过双指标判定的二次抽样, 该算法在设计上兼顾了系统的通信性能和学习性能的提升。

本文设计的用户调度策略旨在通过合理的 UE 关联策略有效避免“落伍者”效应, 实现每轮通信时延与总通信回合数两者之间的权衡, 从而最小化系统总时延。在功率分配方案确定的前提下, 可将其优化问题表述为

$$\begin{aligned}
& \min_{\mathbf{a}} T_c(\mathbf{a}) \\
& \text{s.t. } C_1, C_3 \\
& \quad C_2: \sum_{k \in \mathcal{K}} a_k (1 - a_k) \leq 0 \\
& \quad C_4: 0 \leq a_k \leq 1
\end{aligned} \quad (19)$$

1) 第一次用户抽样。第一次用户抽样旨在提高FL的学习性能,因此选择拥有“优质”训练集的用户。数据集的“优质”与否基于数据集大小和数据集异构程度(方差大小)进行判定。为此,引入指标数据“重要度” $p_{\text{data},k}$ 来衡量 UE_k 持有数据的“优质”与否。具体定义为

$$p_{\text{data},k} = \frac{D_k}{D} \frac{1}{\text{Var}(D_k)} \quad (20)$$

其中, D_k 和 D 分别表示 UE_k 数据集大小和系统总数据集大小, D_k 代表 UE_k 持有的训练数据集, $\text{Var}(\cdot)$ 代表方差大小。

在联邦学习中,数据集大小反映了可用训练集的规模,较大的数据集可以使模型学习到更多的特征和模式,从而避免由于训练数据太小而引起的过拟合问题,进而提高模型的泛化能力。而数据集的方差反映了数据异构的程度,数据异构性高(如不同用户的数据分布差异大)可能导致模型难以收敛,从而需要更多的通信回合来融合模型更新。相对而言,低异构性的数据集意味着数据分布较为一致,有助于模型的快速收敛和泛化,只需较少的通信回合即可达到满意的模型性能,从而降低通信成本。式(20)中, $p_{\text{data},k}$ 越大 UE 被选择的概率就越大。即训练数据集规模大且数据集异构性小的用户会被优先选择。根据 $p_{\text{data},k}$ 选择数据量大且异构程度低的用户参与联邦学习任务可以有效地提升FL模型的性能,减少所需的通信回合,从而实现高效、低成本的分式学习。

2) 二次抽样。侧重考虑用户的信道状态,并引入一个反映用户当前链路状态的指标“中断概率”。

$$p_{\text{out},k} = 2 \left(\frac{T_k^r}{T_{\text{round}}} \right) - 1 \quad (21)$$

其中, T_k^r 是第 r 轮通信 UE_k 的耗时, T_{round} 代表一个FL通信回合允许的最大执行时间。中断概率越小表示用户 k 此时的链路状态越好,也意味着用户 k 能够以更低的时延和更高的可靠性完成通信任务,这对于联邦学习中的模型更新和聚合至关重要。根据式(21)可知,只有能够在规定时间内完成学习任务的用户才有可能被选中,且耗时越少的用户被选入最终参与FL训练过程用户集 $\hat{\mathcal{U}}$ 的概率越大。

综合考虑数据“重要度”和“中断概率”两项指标为用户调度的考量因素,可以同时评估用户对模型训练的数据贡献和通信质量的贡献,从而选出既能够提高学习性能又能高效通信的用户组合,实现系统通信资源和计算资源利用率的平衡。

综上,基于资源优先的二次抽样算法如算法1所示。

算法1 基于资源优先的二次抽样算法

初始化系统网络和FL参数: AP和UE数目 M, K , 功率 p_d, p_u, p_p , 功率控制因子 η, ς , 模型大小 S_d, S_u , UE配置信息 \mathcal{D}_k, L, f, c , 最少参与用户数 N_{th} , 每轮截止时间 T_{round} ; 初始化 $\mathcal{U} = \emptyset, \hat{\mathcal{U}} = \emptyset, a_k = 0$

1) 根据式(20)计算每个 UE_k 的 $p_{\text{data},k}$;

2) 将 $p_{\text{data},k}$ 按降序排列取前 $w \times K$ (w 为抽样率, 本文设置为0.8)个值对应的编号 k ;

3) 更新 $\mathcal{U} = \mathcal{U} \cup \{k\}$;

4) i 从1取到 wK 做如下操作:

5) 取 $k = \mathcal{U} \{i\}$, 并根据式(21)计算 $p_{\text{out},k}$;

6) 在0~1之间产生一个随机数 rd ;

7) if $\text{rd} > p_{\text{out},k}$

8) 设置 $a_k = 1$, 并更新 $\hat{\mathcal{U}} = \hat{\mathcal{U}} \cup \{k\}$;

9) else

10) 设置 $a_k = 0$;

11) end if

12) 输出调度方案 \mathbf{a} 和最终被选择用户集 $\hat{\mathcal{U}}$ 。

2.2 基于二分法的功率分配算法

在用户调度算法中,为提升学习模型的性能,本文优先选择数据“重要度”高的用户。考虑到大数据集会增加通信负担,本文设想通过功率资源分配优化算法,在一定程度上缓解大数据集计算和传输带来的负面影响。为此,本文以最小化系统总时延为目标,对上行和下行的发射功率进行联合优化。针对这样一个随机非凸的问题,将其解耦为2个子问题,并提出了一种BM-PA算法。在该算法中,首先将联合优化问题分解为上行和下行2个子问题。然后,通过二分法迭代求解每个子问题,以逐步逼近最优解。通过动态地调整发射功率的分配,确保每个用户的通信资源与其数据贡献相匹配,从而避免资源占有比例失配引起的通信性和联邦学习性能的恶化,最终实现系统通信资源和计算资源利用率的平衡。

执行RPSS-US算法后得到用户调度方案 \mathbf{a} ,各

个 UE 的计算时延退化为常量, 则功率分配问题可简化为式(22)所示优化模型。

$$\begin{aligned} \min_{\boldsymbol{\eta}, \boldsymbol{\varsigma}} & \left(\max_{k \in \hat{\mathcal{U}}} \frac{S_d}{R_{d,k}(\boldsymbol{\eta})} + \max_{k \in \hat{\mathcal{U}}} \frac{S_u}{R_{u,k}(\boldsymbol{\varsigma})} \right) \\ \text{s.t. } C_1: & \sum_{k \in \mathcal{K}} \sigma_{mk}^2 \eta_{mk} \leq 1 \\ C_2: & 0 \leq \eta_{mk} \\ C_3: & 0 \leq \varsigma_k \leq 1 \end{aligned} \quad (22)$$

2 个不同方向的功率分配方案可以独立进行优化求解。因此, 上述资源分配子问题式(22)可以进一步解耦为以下 2 个问题式(23)和式(24)。

$$\begin{aligned} \text{P1} \quad \min_{\boldsymbol{\eta}} \max_{k \in \hat{\mathcal{U}}} & \frac{S_d}{R_{d,k}(\boldsymbol{\eta})} \\ \text{s.t. } C_1: & \sum_{k \in \hat{\mathcal{U}}} \sigma_{mk}^2 \eta_{mk} \leq 1 \\ C_2: & 0 \leq \eta_{mk} \end{aligned} \quad (23)$$

$$\begin{aligned} \text{P2} \quad \min_{\boldsymbol{\varsigma}} \max_{k \in \hat{\mathcal{U}}} & \frac{S_u}{R_{u,k}(\boldsymbol{\varsigma})} \\ \text{s.t. } C_1: & 0 \leq \varsigma_k \leq 1 \end{aligned} \quad (24)$$

这两个问题属于同一类型的问题, 采用相同的方法求解, 本文采用二分法求解。具体求解细节如下。

1) 下行功率分配子问题 P1。求解 $\min \max S_d/R_{d,k}$ 相当于求解 $\max \min R_{d,k}$ 。则问题式(23)可以改写为问题式(25)。

$$\begin{aligned} \max_{\boldsymbol{\eta}} \min_{k \in \hat{\mathcal{U}}} & \frac{p_d \left(\sum_{k \in \hat{\mathcal{U}}} \sum_{m \in \mathcal{M}} \sqrt{\eta_{mk}} \sigma_{mk}^2 \right)^2}{p_d \sum_{k' \in \hat{\mathcal{U}}} A_{kk'}^2 |\boldsymbol{\phi}_k^H \boldsymbol{\phi}_k|^2 + p_d \sum_{k \in \hat{\mathcal{U}}} B_k + 1} \\ \text{s.t. } & C_1, C_2 \end{aligned} \quad (25)$$

引入松弛变量 $\theta_{k'k}$ 和 \mathcal{G}_m , 于是将问题式(25)改写为

$$\begin{aligned} \max_{v_{mk}, \theta_{k'k}, \mathcal{G}_m} \min_{k \in \hat{\mathcal{U}}} & \frac{\left(\sum_{k \in \hat{\mathcal{U}}} \sum_{m \in \mathcal{M}} v_{mk} \gamma_{mk} \right)^2}{\left(\sum_{k' \in \hat{\mathcal{U}}} |\boldsymbol{\phi}_k^H \boldsymbol{\phi}_k|^2 \theta_{k'k}^2 + \sum_{k \in \hat{\mathcal{U}}} \sum_{m \in \mathcal{M}} \beta_{mk} \mathcal{G}_m^2 + \frac{1}{p_d} \right)} \\ \text{s.t. } C_1: & \sum_{k' \in \hat{\mathcal{U}}} \gamma_{mk} v_{mk'} \leq \mathcal{G}_m^2 \\ C_2: & \sum_{m \in \mathcal{M}} \gamma_{mk} v_{mk'} \frac{\beta_{mk}}{\beta_{mk'}} \leq \theta_{k'k}, \forall k' \neq k \\ C_3: & 0 \leq \mathcal{G}_m \leq 1 \\ C_4: & 0 \leq v_{mk} \end{aligned} \quad (26)$$

其中, $v_{mk} \triangleq \sqrt{\eta_{mk}}$, $\gamma_{mk} \triangleq \sigma_{mk}^2$ 。

2) 上行功率分配子问题 P2。类似于 P1, 问题式(24)等价于

$$\begin{aligned} \max_{\boldsymbol{\varsigma}} \min_{k \in \hat{\mathcal{U}}} & \frac{\varsigma_k \left(\sum_{m \in \mathcal{M}} \sigma_{mk}^2 \gamma_{mk} \right)^2}{\sum_{k' \in \hat{\mathcal{U}}} U_{kk'}^2 |\boldsymbol{\phi}_k^H \boldsymbol{\phi}_k|^2 + \sum_{k \in \hat{\mathcal{U}}} V_k + \frac{W_k}{p_u}} \\ \text{s.t. } & C_1 \end{aligned} \quad (27)$$

问题式(26)和问题式(27)可以通过二分搜索^[22]在每一步中求解一系列凸可行性问题, 从而有效地求解。对于下行功率分配, 相当于求解

$$\begin{cases} \|\mathbf{Y}_k\| \leq \frac{1}{\sqrt{t}} \sum_{m \in \mathcal{M}} \gamma_{mk} v_{mk} \\ \sum_{k \in \hat{\mathcal{U}}} \gamma_{mk} v_{mk}^2 \leq \mathcal{G}_m^2 \\ \sum_{m \in \mathcal{M}} \gamma_{mk'} \frac{\beta_{mk}}{\beta_{mk'}} v_{mk'} \leq \theta_{k'k}, \forall k' \neq k \\ 0 \leq \mathcal{G}_m \leq 1 \\ v_{mk} \geq 0 \end{cases} \quad (28)$$

其中, $\mathbf{Y}_k \triangleq \left[\mathbf{Y}_1^T \mathbf{I}_{-k} \mathbf{Y}_2^T \frac{1}{\sqrt{p_d}} \right]^T$, $\mathbf{Y}_1 \triangleq [\boldsymbol{\phi}_1^H \boldsymbol{\phi}_k \theta_{1k}, \dots, \boldsymbol{\phi}_K^H \boldsymbol{\phi}_k \theta_{Kk}]^H$, \mathbf{I}_{-k} 是 $K \times K$ 的单位矩阵去掉第 k 列后大小为 $K \times (K-1)$ 的矩阵, 并且 $\mathbf{Y}_2 \triangleq [\sqrt{\beta_{1k}} \mathcal{G}_1, \dots, \sqrt{\beta_{Mk}} \mathcal{G}_M]^T$ 。

类似地, 求解上行功率分配相当于求解

$$\begin{cases} \|\mathbf{Z}_k\| \leq \frac{1}{\sqrt{t}} \varsigma_k \sum_{m \in \mathcal{M}} \gamma_{mk} \\ 0 \leq \varsigma_k \leq 1 \end{cases} \quad (29)$$

其中, $\mathbf{Z}_k \triangleq [\mathbf{Z}_1^T \mathbf{I}_{-k} \mathbf{Z}_2^T \mathbf{Z}_3^T]^T$, $\mathbf{Z}_1 \triangleq [\boldsymbol{\phi}_1^H \boldsymbol{\phi}_k \varsigma_k \mathcal{G}_{1k}, \dots, \boldsymbol{\phi}_K^H \boldsymbol{\phi}_k \varsigma_k \mathcal{G}_{Kk}]^T$, $\mathcal{G}_{k'k} = \sum_{m \in \mathcal{M}} \frac{\gamma_{mk} \beta_{mk'}}{\beta_{mk}}$, $\mathbf{Z}_3 \triangleq \left[\sqrt{\frac{\gamma_{1k}}{p_u}}, \dots, \sqrt{\frac{\gamma_{Mk}}{p_u}} \right]^T$, $\mathbf{Z}_2 \triangleq \left[\sqrt{\beta_{1k} \gamma_{1k}} \varsigma_k, \dots, \sqrt{\beta_{Mk} \gamma_{Mk}} \varsigma_k \right]^T$ 。

具体的基于二分法的功率分配算法如算法 2 所示。

算法 2 基于二分法的功率分配算法

初始化网络系统基本参数: AP 和 UE 数目 M, K , 功率 p_d, p_u, p_p , 用户调度策略 \mathbf{a} ; 设置迭代精度 $\varepsilon = 0.001$; 设置 t_{\min} 和 t_{\max}

1) 循环;

- 2) 更新 $t = \frac{t_{\min} + t_{\max}}{2}$;
- 3) if case1:下行功率分配
- 4) 求解凸问题式(28);
- 5) if 凸问题式(28)具有可行解集 \mathcal{F}
- 6) 更新 $t_{\min} = t$;
- 7) else if 凸问题式(28)不可解
- 8) 更新 $t_{\max} = t$;
- 9) end if
- 10) else if case2:上行功率分配
- 11) 求解凸问题式(29);
- 12) if 凸问题式(29)具有可行解集 $\tilde{\mathcal{F}}$
- 13) 更新 $t_{\min} = t$;
- 14) else if 凸问题式(29)不可解
- 15) 更新 $t_{\max} = t$;
- 16) end if
- 17) end if
- 18) 直到 $t_{\max} - t_{\min} \leq \varepsilon$
- 19) if case1: 下行功率分配

$$20) \text{ 最优解 } \boldsymbol{\eta}^* = \arg \min_{\boldsymbol{\eta} \in \mathcal{F}} \left(\max_{k \in \mathcal{U}} \frac{S_d}{R_{d,k}(\boldsymbol{\eta})} \right);$$

- 21) else if case2: 上行功率分配

$$22) \text{ 最优解 } \boldsymbol{\zeta}^* = \arg \min_{\boldsymbol{\zeta} \in \tilde{\mathcal{F}}} \left(\max_{k \in \mathcal{U}} \frac{S_u}{R_{u,k}(\boldsymbol{\zeta})} \right);$$

- 23) end if

- 24) 输出最优的上下行功率分配方案 $\boldsymbol{\zeta}^*$ 和 $\boldsymbol{\eta}^*$ 。

2.3 基于交替迭代优化的联合优化算法

本文采用交替迭代优化方法解决联合优化问题。具体而言,首先,在大尺度相干时间内(即一个通信回合 r 内),对于给定的随机用户选择策略 \boldsymbol{a} ,资源分配子问题式(22)通过算法2(BM-PA算法)进行若干次迭代求解,以获得最优解 $\boldsymbol{\zeta}^*$ 和 $\boldsymbol{\eta}^*$ 。随后,将 $\boldsymbol{\zeta}^*, \boldsymbol{\eta}^*$ 作为算法1(RPSS-US算法)功率控制因子初始化的值,执行RPSS-US算法获得最优调度方案 \boldsymbol{a}^* ,并持续更新 $\boldsymbol{a}^{r+1} = \boldsymbol{a}^*, \boldsymbol{\zeta}^{r+1} = \boldsymbol{\zeta}^*, \boldsymbol{\eta}^{r+1} = \boldsymbol{\eta}^*$ 直至达到最大迭代次数。基于交替迭代优化的联合优化算法如算法3所示。

算法3 基于交替迭代优化的联合优化算法

初始化网络系统和FL相关的基本参数;初始化首轮调度策略 \boldsymbol{a}^0 ; 总迭代次数 R ; 设置 $r = 1$

- 1) 循环;

- 2) 执行算法2(BM-PA算法),求解功率分配子问题得到最优功率分配方案 $\boldsymbol{\eta}^*$ 和 $\boldsymbol{\zeta}^*$;

- 3) 更新 $\boldsymbol{\eta}^{r+1} = \boldsymbol{\eta}^*$;

- 4) 更新 $\boldsymbol{\zeta}^{r+1} = \boldsymbol{\zeta}^*$;

- 5) 把 $\boldsymbol{\eta}^*, \boldsymbol{\zeta}^*$ 作为算法1(RPSS-US算法)功率控制因子的初始值,执行算法得到最优用户调度策略 \boldsymbol{a}^*

- 6) 更新 $\boldsymbol{a}^{r+1} = \boldsymbol{a}^*$;

- 7) 更新 $r = r + 1$;

- 8) 直到 $r > R$

- 9) 输出最优的用户调度方案 \boldsymbol{a} 以及最优的上下行功率分配方案 $\boldsymbol{\zeta}^*$ 和 $\boldsymbol{\eta}^*$ 。

2.4 计算复杂度分析

本文算法1(RPSS-US算法)第一次采样需要遍历用户寻找学习性能较优的用户,算法复杂度为 $O(K)$,第二次抽样需要遍历首次抽样的用户集,算法复杂度为 $O(wK)$,则用户调度算法的复杂度为 $O((1+w)K)$,其中 w 为首次用户采样的比例, K 为系统总用户数目。本文算法2(BM-PA算法)在单个方向上采用二分法求解,该算法最大的求解复杂度为 $O(\text{lb}(t_{\max} - t_{\min}))$,假使上行和下行的功率分配设置相同的 t_{\max} 和 t_{\min} ,则执行BM-PA算法的复杂度为 $O(2\text{lb}(t_{\max} - t_{\min}))$ 。本文算法3(基于交替迭代优化的联合优化算法)执行 R 次迭代,复杂度为 $O(R((1+w)K + 2\text{lb}(t_{\max} - t_{\min})))$ 。文献[4]在进行功率分配算法之前还需要额外使用算法复杂度为 $O(MK)$ 的贪婪算法进行导频的分配,其中 M 为AP总数目,然后执行算法复杂度同样为 $O(2\text{lb}(t_{\max} - t_{\min}))$ 的功率分配算法,经过 R 次迭代后其算法总体复杂度为 $O(R(MK + 2\text{lb}(t_{\max} - t_{\min})))$ 。此外,文献[9]需要通过求解一系列的二阶锥规划问题获得功率分配和用户选择方案,其中一个二阶锥规划问题的求解所需迭代次数为 $\sqrt{3(K+M)+MK}$,每次求解的复杂度大约为 $O(n((M+1)K^3 + 2nK^2 + 2M^2))$, n 表示需要优化的总变量的个数, R 次迭代后其算法复杂度为 $O(nR\sqrt{3(K+M)+MK}((M+1)K^3 + 2nK^2 + 2M^2))$ 。相比之下,本文所提算法的复杂度较低。

3 仿真分析

为了验证本文算法的有效性,本文利用MAT-

LAB 机器学习工具箱进行了 FL 算法的手写体数字识别仿真，并对所提出的 RPSS-US 算法和 BM-PA 算法的有效性进行了验证。在仿真实验中，本文假设有 100 个 AP 和 20 个 UE 随机均匀分布在一个 500 m × 500 m 的特定区域内。网络系统其他参数具体配置如表 2 所示，其中，一个 sample 指一个训练样本的大小。

表 2 系统参数配置

参数及其含义	取值
导频、下行和上行归一化噪声功率 $p_p, p_d, p_u/W$	0.2、1、0.2
相干时间和导频序列长度 $\tau_c, \tau_p/sample$	200、20
下发和上传数据量的大小 $S_d, S_u/MB$	5
系统上、下行传输带宽 $B^U, B^D/MHz$	1、20
噪声功率 σ_0^2/dBm	-92
第 k 个 UE 训练数据集大小 $D_k/sample$	$1 \times 10^4 \sim 40 \times 10^4$
用户设备 CPU 处理频率 f	100 MHz~2 GHz
最少参与用户数目 $N_{th}/个$	5
FL 训练任务相关参数 q	90
UE 本地模型更新所需迭代次数 $L/次$	5

本文将所提算法 3（以下用 PROPOSED 表示）与以下 4 种对比方案进行了对比验证。

对比方案 1（以下用 NOUS MMPA 表示）：未进行用户调度，采用文献[4]改进的 Max-Min 功率分配方案。

对比方案 2（以下用 NOUS BMPA 表示）：未进行用户调度，功率分配采用本文提出的 BM-PA 算法。

对比方案 3（以下用 PRBS EEPa 表示）：采用文献[9]提出的基于接收功率选择（PRBS, received power based selection）的用户调度算法和能效最大化的功率分配方案。

对比方案 4（以下用 CPCS AIEO）：采用文献[16]提出的基于信道排序的信道较用户优先（CPCS, channel poor user priority based on channel sorting）接入的调度策略，并采用以能耗优化为目标的交替迭代（AIEO, alternating iterations with the goal of energy consumption optimization）算法进行功率分配。

图 4 展示了在执行 300 次 FL 迭代后，5 种不同优化方案实现的系统下行吞吐量。

图 5 则展示了 5 种不同优化方案实现的系统下行吞吐量的累积分布函数曲线。

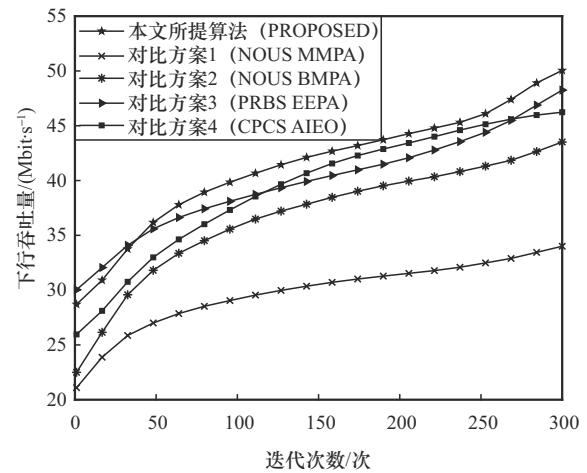


图 4 5 种不同优化方案实现的系统下行吞吐量

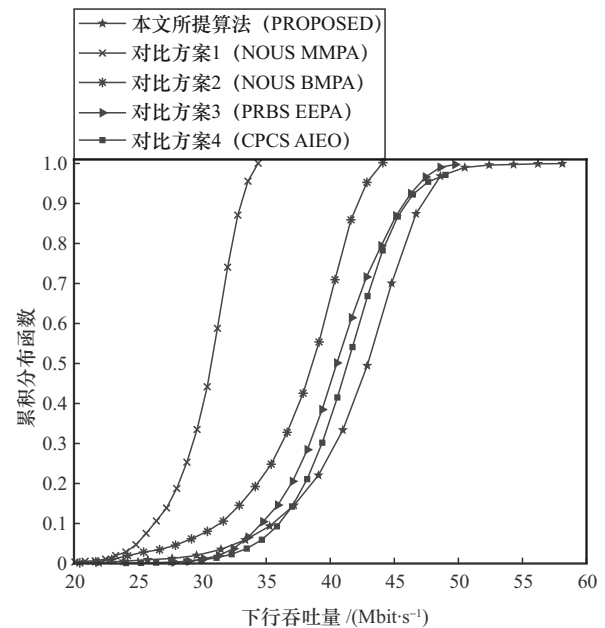


图 5 5 种不同优化方案实现的系统下行吞吐量的累积分布函数曲线

根据图 4 结果显示，PROPOSED 在提升系统下行吞吐量方面表现最佳。具体数值表明，相较于对比方案，所提方案能够在其基础上最大额外提升下行吞吐量分别达到 47.19%、27.57%、5.50% 和 9.55%，平均则可实现下行吞吐量分别提升 38.70%、13.28%、3.32% 和 5.14%。对比结果还显示，同样在没有用户调度的情况下，本文提出的 NOUS BMPA 算法（对比方案 2）相较于文献[4]中改进的 NOUS MMPA 功率分配算法（对比方案 1），下行吞吐量平均提升了 22.60%。进一步对比所提方案与对比方案 2（无用户调度）可发现，将 RPSS-US 算法与 BM-PA 算法结合使用，相较于单独使用 BM-PA 算法，系统容量提升效果更显著，数值结果表明这

种提升幅度大约为13.28%，这是因为本文提出的RPSS-US算法在设计时降低了“落伍者”参与FL过程的概率，使得原本可能浪费在“落伍者”的资源可以重新分配给参与FL过程的用户，从而进一步提升功率分配的效果。与此同时，相较于文献[9]的PRBS EEPA算法（对比方案3）和文献[16]的CPCS AIEO算法（对比方案4）这些在用户调度时只考虑用户信道状态的算法，本文所提资源优化算法在提升系统下行吞吐量上也具有明显优势，这种优势分别为3.32%和5.14%。图5显示，使用本文所提算法进行资源优化后，系统下行吞吐量平均集中在41~45 Mbit/s，而性能更接近本文所提算法的PRBS EEPA算法和CPCS AIEO算法优化后系统的下行吞吐量分别集中在37~42 Mbit/s和39~43 Mbit/s。这得益于所提算法平衡了用户通信和计算资源的利用，降低了链路质量较差的用户参与FL学习的概率，从而提高了整体系统的吞吐量。

图6展示了在执行300次FL迭代后，5种不同优化方案在系统上行吞吐量方面的表现。

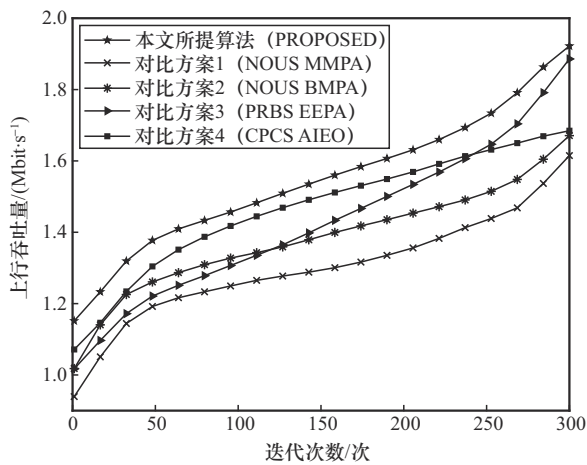


图6 5种不同优化方案实现的系统上行吞吐量

图7则给出了5种不同优化方案实现的系统上行吞吐量的累积分布函数曲线。

根据图6的结果显示，所提算法（PROPOSED）在提升系统上行吞吐量方面效果也最佳。仿真结果表明，相较于对比方案，所提算法能够额外提升上行吞吐量最大分别达到22.60%、16.10%、13.21%和12.36%，平均则可提升18.87%、11.73%、8.82%和5.15%。对比本文提出的NOUS BMPA算法（对比方案2）与文献[4]改进的NOUS MMPA功率分配算法（对比方案1），在不采取用

户调度的情况下，NOUS BMPA算法相较于NOUS MMPA功率分配算法，可额外平均提升系统上行吞吐量6.40%。此外，本文所提方案在提升系统上行容量方面相较于PRBS EEPA算法和CPCS AIEO算法也表现优异，上行吞吐量平均可额外提升8.82%和5.15%。与下行方向的情况类似，将RPSS-US算法与BM-PA算法结合使用，比单独使用BM-PA算法系统容量提升了11.73%。这是因为本文在进行用户调度算法设计的时候赋予链路质量好、资源丰富的“优质用户”更高的优先级来参与FL过程，这些用户对系统性能提升的贡献更大。图7结果显示，使用本文提出的算法（PROPOSED）优化后，系统上行吞吐量显著高于其他4种对比方案，平均集中在1.5~2 Mbit/s。

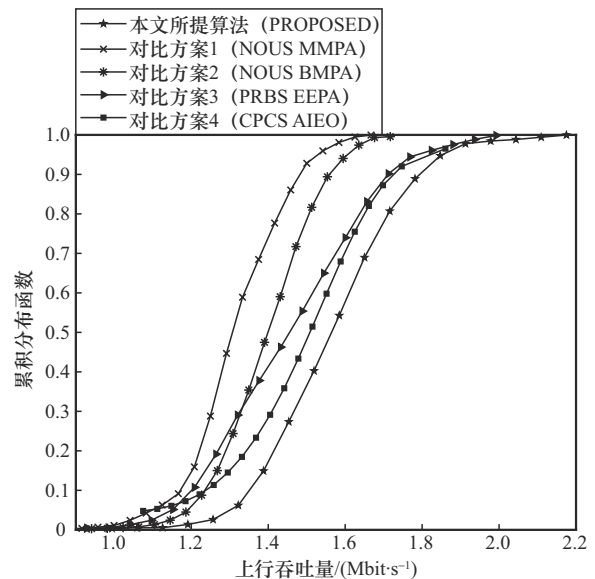


图7 不同优化方案实现的系统上行吞吐量的累积分布函数曲线

结合图4~图7，本文所提算法相较于基于接收功率的调度和基于信道排序的信道较差用户优先接入的调度，在系统容量提升方面更具有优势。这是因为本文的调度方案在设计时，不仅考虑了用户的信道状态，还额外考量了用户的计算资源。具体来说，本文算法通过综合评估用户的计算资源和当前链路负载情况，动态调整调度策略，从而更有效地分配网络资源，提升整体系统性能。然而，由图5和图7可知，本文所提算法在速率变化率上效果稍逊，就CPCS AIEO算法而言，基于信道排序的调度策略给信道状态较差的用户较高的接入优先级，可以有效降低系统中用户速率的极端波动，维持整体

系统速率的均衡性。虽然该方案未能充分利用计算资源，导致系统容量优化效果不如本文所提算法，但其优化后系统用户速率变化的差异较小，表现出更好的均衡性。

为了方便度量时间节省效果，本文定义一个时间节省指标： $\frac{\text{长耗时} - \text{短耗时}}{\text{长耗时}} \times 100\%$ 。

图 8 展示了在 FL 训练任务最大迭代次数为 150 时，每个通信回合不同优化方案所需的时间消耗量情况。

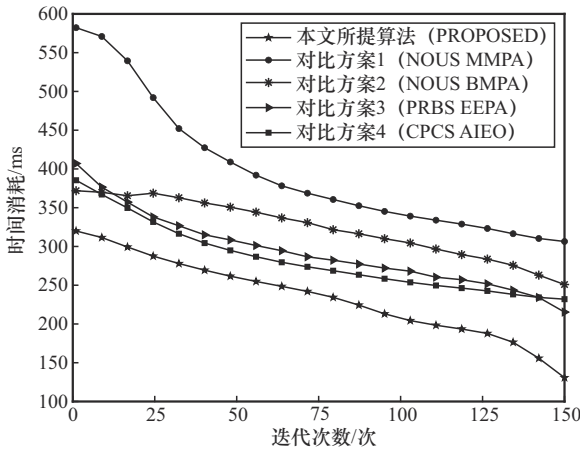


图 8 不同优化方案一个 FL 通信回合的时间消耗

根据图 8 结果显示，与其他 4 种对比方案相比，本文所提算法 (PROPOSED) 在节省时间消耗方面表现最佳。随着迭代次数增加，系统时间消耗呈下降趋势。这是因为随着 FL 的进行，用户的本地模型逐渐接近最优全局模型，算法优化效果逐步增强。具体数据显示，相较于 NOUS MMPA 算法，所提出的算法最大可节省时间 57.33%，平均节省 40.81%。在无用户调度的情况下，本文提出的 NOUS BMPPA 算法相较于文献[4]改进的 NOUS MMPA 功率分配算法平均节省时间消耗为 16.94%。同时进行用户调度和资源分配时，本文所提算法在 PRBS EEPA 算法的基础上平均能节省 21.52% 的时间开销，在 CPCS AIEO 算法的基础上则能平均节省时间 43.65%。这表明了所提算法在降低系统时延方面具有显著的效果。这是因为本文的算法设计重点考虑了系统的时间消耗，因此在时间开销上具有优越性；进一步对比所提算法和对比方案 2，数值结果显示，相比于直接使用 BM-PA 算法对系统进行优化，将其与

RPSS-US 算法结合后时间优化的效果会更好，这种时间优化效果的提升最大可达 47.89%，平均也在 28.23% 左右。这是因为本文在设计 RPSS-US 算法时同时考虑了用户计算资源和通信资源的丰富程度，使得被关联的用户在模型下载、模型上传和模型更新方面的耗时相对较少。图 8 的曲线趋势还表明，随着 FL 逐渐接近尾声，有用户调度的资源分配方案 (对比方案 3 和 PROPOSED) 比无用户调度的方案 (对比方案 1 和 2) 节省时间的效果更为明显。这是由于随着 FL 的进行，数据集的质量对模型的影响越来越大，用户的选择变得更加重要。然而，对于 CPCS AIEO 算法而言，由于其更关注系统能耗的优化，所以后期在时延效果上的提升趋于饱和。图 9 和表 3 给出了不同优化方案在 FL 模型精度分别达到 85% 和 90% 时的时间消耗情况。本文所提出的算法 3 与其他 4 种方案进行了对比。其中，对比方案 2~4 依旧采用上述方案，另外，为了验证本文用户调度算法的有效性，原来的对比方案 1 更改为以下方案。

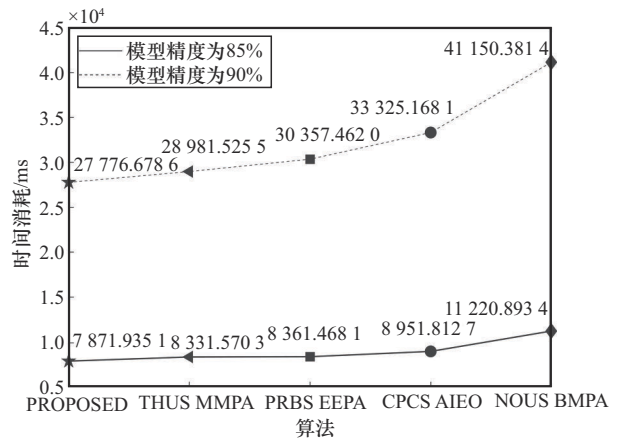


图 9 不同优化方案 FL 模型精度达到 85% 和 90% 的时间消耗

表 3 不同优化方案 FL 模型精度达到 85% 和 90% 的时间消耗

优化方案	时间消耗/ms	
	模型精度为 85%	模型精度为 90%
PROPOSED	7 871.935 1	27 776.678 6
THUS MMPA	8 331.570 3	28 981.525 5
PRBS EEPA	8 361.468 1	30 357.462 0
CPCS AIEO	8 951.812 7	33 325.168 1
NOUS BMPPA	11 220.893 4	41 150.381 4

对比方案1 (THUS MMPA): 采用门限截止的用户调度方案, 即只选择能在规定限制时间内完成学习任务的UE, 并采用Max-Min功率分配方案。

根据图9的结果显示, 无论是模型精度达到85%还是90%, 所提出的算法所需的时间消耗都明显少于其他对比方案。当模型精度为85%时, 提出的算法仅需约7 872 ms (约7.8 s), 相比于THUS MMPA算法 (约8.3 s)、PRBS EEPA算法 (约8.3 s)、CPCS AIEO算法 (约8.9 s) 以及NOUS BMPA算法 (约11.2 s), 分别节省了约5.52%、5.85%、12.06%和29.85%的时间。当模型精度达到90%时, 提出的方案所需的时间约为27.8 s, 相比于THUS MMPA算法 (约29 s)、PRBS EEPA算法 (约30.4 s)、CPCS AIEO算法 (约33.3 s) 以及NOUS BMPA算法 (约41.2 s), 分别节省了约4.16%、8.50%、16.65%和32.50%的时间。此外, 模型精度从85%提升到90%时, 所提算法所需的额外时间消耗相对于其他对比方案方法也较少, 这是因为RPSS-US算法考虑了用户的“数据重要度”, 这对于提升FL性能至关重要。该算法选择了对FL性能提升贡献大的用户。而且相比于没有用户调度策略的方案 (对比方案2), 进行用户调度的方案 (本文所提算法、对比方案1、3和4) 都能在一定程度上优化FL的收敛速率, 其中本文所提算法的效果最佳。这进一步验证了本文所提算法在时间消耗方面的优越性和有效性。图10和表4展示了在采用FL框架和不采用FL框架的CF mMIMO系统中, 不同总UE数目下完成一次通信的平均时延对比情况。

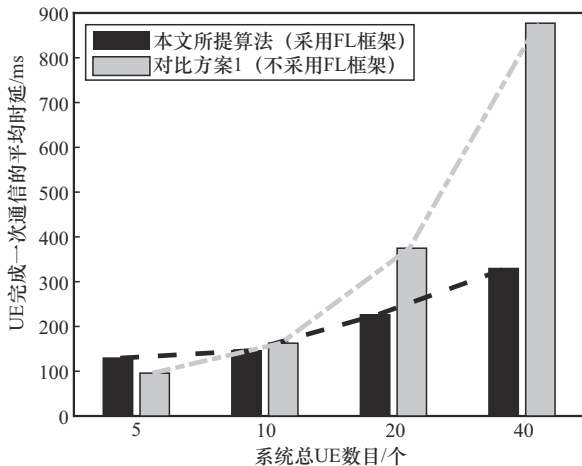


图10 采用FL框架与不采用FL框架的UE平均时延对比

表4 采用FL框架与不采用FL框架的UE平均通信时延对比

系统总UE数目/个	本文所提算法 (采用FL框架)/ms	对比方案1 (不采用FL框架)/ms
5	129.349 3	95.817 3
10	145.774 4	162.638 1
20	226.162 9	374.741 9
40	329.401 6	877.176 0

结果表明, 在系统低负载的情况下 (5个UE), 不采用FL框架通信时延较少。在低负载情况下, 系统中参与通信的用户设备 (UE) 数量少, 直接进行集中式的模型训练或数据传输时, 通信开销较低。每个UE与集中式服务器之间的通信频次和数据量相对较小, 因此整体的通信时延也较少。相比之下, FL训练过程需要多个UE和中央服务器之间进行频繁的参数交换。这包括多轮的模型更新、传输和聚合, 增加了通信开销。然而, 当网络负载上升时 (10个UE、20个UE、40个UE), 不采用FL框架的时间开销急剧上升。因为在FL框架下, 各个UE只需传输训练后的模型参数而非大量的原始数据。这种方式显著降低了数据传输量, 尤其是在用户数量较多时, 更能有效减少通信时延。相比之下, 不采用FL框架的方法需要传输大量的原始数据到中央服务器处理, 随着用户数量增加, 通信时延急剧增加。这种对比结果表明, 在低负载情况下, 不采用FL框架的集中式通信方法更为高效。然而, 在高负载情况下, FL框架通过减少数据传输量, 在通信时延方面表现出明显的优势。因此, 在高负载环境中, 采用FL框架的CF mMIMO系统在优化通信时延方面具有更好的效果。

4 结束语

本文将联邦学习引入无蜂窝网络系统, 设计了一个联合优化问题, 涉及UE选择和上下行功率分配, 以最小化无蜂窝网络中FL过程的执行时间。针对用户调度问题, 提出了一种基于资源优先的二次抽样算法, 通过综合考虑用户对全局模型训练和系统容量的贡献大小进行选择。对于系统通信时延问题, 建立了联邦学习框架下的时间消耗分析模型, 并构建时延最小化的目标函数表达式, 提出一种基于二分法的功率分配算法。

实验仿真结果表明,相较于对比方案,本文提出的算法显著减少了FL训练时间,加快了模型的收敛速度,最好的情况下可以节省57.33%的时间开销。除此之外,本文所提算法在提升系统容量方面也取得了明显效果。具体而言,在下行吞吐量方面最大可提升47.19%,上行吞吐量则最大可提升了22.60%。综上所述,本文所提算法在有效分配资源和优化性能方面表现优异,显著提升了系统的整体效率和性能,使得无蜂窝网络中的FL过程更加高效和可靠。然而,本文的研究重点集中在对于通信时延的考虑,尚未将系统的能耗考虑在内。本文算法在系统速率的变化率方面也不佳,因此如何提升系统用户速率的均衡性以及系统时延与能耗的权衡将会成为下一步的研究方向。

参考文献:

- [1] LI T, SAHU A K, TALWALKAR A, et al. Federated learning: challenges, methods, and future directions[J]. *IEEE Signal Processing Magazine*, 2020, 37(3): 50-60.
- [2] ELHOUSHY S, IBRAHIM M, HAMOUDA W. Cell-free massive MIMO: a survey[J]. *IEEE Communications Surveys & Tutorials*, 2022, 24(1): 492-523.
- [3] KASSAM J, CASTANHEIRA D, SILVA A, et al. A review on cell-free massive MIMO systems[J]. *Electronics*, 2023, 12(4): 1001.
- [4] NGO H Q, ASHIKHMIN A, YANG H, et al. Cell-free massive MIMO versus small cells[J]. *IEEE Transactions on Wireless Communications*, 2017, 16(3): 1834-1850.
- [5] VU T T, NGO D T, TRAN N H, et al. Cell-free massive MIMO for wireless federated learning[J]. *IEEE Transactions on Wireless Communications*, 2020, 19(10): 6377-6392.
- [6] CHEN M Z, GÜNDÜZ D, HUANG K B, et al. Distributed learning in wireless networks: recent progress and future challenges[J]. *IEEE Journal on Selected Areas in Communications*, 2021, 39(12): 3579-3605.
- [7] XIA W C, QUEK T Q S, GUO K, et al. Multi-armed bandit-based client scheduling for federated learning[J]. *IEEE Transactions on Wireless Communications*, 2020, 19(11): 7108-7123.
- [8] ZAEEM R M, DUNCAN J C M, MARTINS W A, et al. Resource allocation and user scheduling design for user-centric cell-free massive MIMO systems[C]//*Proceedings of the 2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Piscataway: IEEE Press, 2023: 1-6.
- [9] NGO H Q, TRAN L N, DUONG T Q, et al. On the total energy efficiency of cell-free massive MIMO[J]. *IEEE Transactions on Green Communications and Networking*, 2018, 2(1): 25-39.
- [10] 申敏, 裘德市. 无蜂窝大规模MIMO系统接入点动态选择算法[J]. *电讯技术*, 2024, 64(3): 416-422.
SHEN M, QIU D S. A dynamic access point selection algorithm in cell-free massive MIMO systems[J]. *Telecommunication Engineering*, 2024, 64(3): 416-422.
- [11] CHEN M Z, POOR H V, SAAD W, et al. Convergence time minimization of federated learning over wireless networks[C]//*Proceedings of the ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. Piscataway: IEEE Press, 2020: 1-6.
- [12] SHI W Q, ZHOU S, NIU Z S. Device scheduling with fast convergence for wireless federated learning[C]//*Proceedings of the ICC 2020-2020 IEEE International Conference on Communications (ICC)*. Piscataway: IEEE Press, 2020: 1-6.
- [13] CHEN M Z, YANG Z H, SAAD W, et al. A joint learning and communications framework for federated learning over wireless networks[J]. *IEEE Transactions on Wireless Communications*, 2021, 20(1): 269-283.
- [14] VU T T, NGO H Q, NGO D T, et al. Energy-efficient massive MIMO for serving multiple federated learning groups[C]//*Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM)*. Piscataway: IEEE Press, 2021: 1-6.
- [15] MAI T C, NGO H Q, TRAN L N. Energy efficiency maximization in large-scale cell-free massive MIMO: a projected gradient approach[J]. *IEEE Transactions on Wireless Communications*, 2022, 21(8): 6357-6371.
- [16] 姚媛媛, 刘忆秋, 黄赛, 等. 无蜂窝大规模MIMO网络下基于联邦学习的用户接入策略及能耗优化[J]. *通信学报*, 2023, 44(10): 112-123.
YAO Y Y, LIU Y Q, HUANG S, et al. Federated learning-based user access strategy and energy consumption optimization in cell-free massive MIMO network[J]. *Journal on Communications*, 2023, 44(10): 112-123.
- [17] VU T T, NGO D T, NGO H Q, et al. Joint resource allocation to minimize execution time of federated learning in cell-free massive MIMO[J]. *IEEE Internet of Things Journal*, 2022, 9(21): 21736-21750.
- [18] BAO H, GAO B, XIONG K, et al. Joint user cooperation and scheduling for federated learning in CFmMIMO networks[C]//*Proceedings of the 2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*. Piscataway: IEEE Press, 2023: 1-5.
- [19] ALEDHARI M, RAZZAK R, PARIZI R M, et al. Federated learning: a survey on enabling technologies, protocols, and applications[J]. *IEEE Access: Practical Innovations, Open Solutions*, 2020, 8: 140699-140725.
- [20] SUN T, LI D S, WANG B. Decentralized federated averaging[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023, 45(4): 4289-4301.
- [21] LI X, HUANG K, YANG W, et al. On the convergence of FedAvg on non-iid data[J]. *arXiv Preprint, arXiv:1907.02189*, 2019.

- [22] KOOLIVAND M, BAHONAR M H, FAZEL M S. Improving energy efficiency of massive MIMO relay systems using power bisection allocation for cell-edge users[C]//Proceedings of the 2019 27th Iranian Conference on Electrical Engineering (ICEE). Piscataway: IEEE Press, 2019: 1470-1475.

[作者简介]



王华华 (1981-), 男, 山西临汾人, 重庆邮电大学正高级工程师、硕士生导师, 主要研究方向为嵌入式系统、移动通信系统软件开发、基带信号处理、物理层协议。



黄烨霞 (2000-), 女, 湖南邵阳人, 重庆邮电大学硕士生, 主要研究方向为移动通信物理层协议算法与无线网络资源优化。



李玲 (2000-), 女, 重庆人, 重庆邮电大学硕士生, 主要研究方向为移动通信物理层协议算法与边缘卸载。