云边端架构下边缘智能计算关键问题综述：计算优化与计算卸载

董裕民[①②③] 张 静[①②] 谢昌佐[①②③] 李子扬*[①②]

[①](中国科学院空天信息创新研究院 北京 100094)

[②](中国科学院定量遥感信息技术重点实验室 北京 100094)

[③](中国科学院大学电子电气与通信工程学院 北京 100049)

摘 要：近年来，随着入网设备数量与数据体量的急剧增加，以云计算为代表的中心式计算模式的缺点越来越显露出来。边缘计算，即让计算尽量靠近数据源，以减少数据传输时间和网络延迟，作为云计算的补充，已经成为学术界和工业界关注的焦点。该文面向边缘计算中应用较广的实例架构——云边端架构，以及边缘计算的典型应用——边缘智能计算，讨论云边端架构下边缘智能计算的两大关键问题：计算优化和计算卸载。首先分析和梳理了云边端架构下边缘智能计算优化的应用与研究现状。然后讨论了云边端架构下计算卸载的研究思路和现状。最后，总结提出了目前云边端架构下边缘智能计算业务所面临的挑战和未来研究趋势。

关键词：云边端架构；边缘智能计算；计算优化；计算卸载

中图分类号：TN919.1; TP393.09 **文献标识码**：A **文章编号**：1009-5896(2024)03-0765-12

DOI: 10.11999/JEIT230390

A Survey of Key Issues in Edge Intelligent Computing Under Cloud-Edge-Terminal Architecture: Computing Optimization and Computing Offloading

DONG Yumin[①②③] ZHANG Jing[①②] XIE Changzuo[①②③] LI Ziyang[①②]

[①](*Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing* 100094, *China*)

[②](*Key Laboratory of Quantitative Remote Sensing Information Technology, Chinese Academy of Sciences, Beijing* 100094, *China*)

[③](*School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: With the rapid increase in the number of network access devices and the volume of network access data currently, the shortcomings of the centralized computing architecture represented by cloud computing are increasingly exposed. Edge computing, that is making computing as close to the data source as possible to reduce data transmission time and network delay, has become the focus of academia and industry as a supplement to cloud computing. An instance architecture widely used in edge computing: Cloud-Edge-Terminal architecture, and a typical application of edge computing: edge intelligent computing is focused on in this paper. Two key issues of edge intelligent computing under Cloud-Edge-Terminal architecture: computing optimization and computing offloading is analyzed. First, the research focus of edge intelligent computing is analyzed, and the application and research status of intelligent computing optimization under Cloud-Edge-Terminal architecture is combed. Then the research ideas and current situation of computing offloading under Cloud-Edge-Terminal architecture is discussed. Finally, the challenges and research trends of edge intelligent computing under Cloud-Edge-Terminal architecture is summarized.

Key words: Cloud-Edge-Terminal; Edge intelligent computing; Computing optimization; Computing offloading

收稿日期：2023-05-09；改回日期：2023-07-07；网络出版：2023-07-14

*通信作者： 李子扬 zyli@aircas.ac.cn

基金项目：国家重点研发计划 (2021YFC3002204)

Foundation Item: National Key R&D Program (2021YFC3002204)

1 引言

随着物联网技术的飞速发展，越来越多的终端设备被接入网络。截至2022年底，我国3家基础电信企业发展蜂窝物联网用户1.8×10^9户，比上年增加4.5×10^8户[1]。伴随终端数量的急速增加，目前主流的云计算体系结构暴露出了越来越多的不足。

首先是隐私安全问题，终端设备产生了大量的隐私数据，这些数据通过网络传输到云服务器上进行处理，可能造成用户隐私的泄露[2]。其次，由于接入网络的设备数量持续高速增长，需要减轻云服务器的计算和传输压力[3]。对于时间敏感型的业务，如自动驾驶[4]、虚拟现实[5]等，若将全部数据传输至云服务器进行计算可能会带来不能接受的时延[6]。最后，数据中心及其计算服务已经成为全球能源消耗的主力之一，如果单纯使用云服务器进行集中式计算，设备端的计算资源会被严重浪费，为云服务器带来更多压力从而增加成本与碳排放[7]。

为解决上述问题，学界提出了边缘计算框架，让计算尽量靠近终端设备[8]，以提供低延迟、高可靠、高安全的服务。与云计算不同，边缘计算的计算过程发生在云服务器和终端设备路径之间[9]，由终端设备、边缘服务器、云服务器共同完成计算业务，边缘设备同时作为数据的提供者和使用者，参与计算并接收计算结果。

边缘服务器，是边缘计算的重要概念。这一概念出现于内容分发网络[10]，起初是为了解决主干网的拥堵，负责进行常用的高访问的静态文件的缓存以便于实现附近用户请求的快速分发。该方法有效减轻了网络传输的负担[11]。随着网络技术的发展，边缘服务器的作用也从静态文件分发扩展到了各类计算服务，边缘服务器与用户(终端)地理位置相近，作为云服务器的补充为用户提供低时延的服务。

将云服务器、边缘服务器、终端设备结合起来的体系结构称为云边端架构，是边缘计算的一个子集实例。目前，云边端架构已经广泛应用于智慧城市[12]、自动驾驶[13]、工业控制[14]、智慧农业[15]、星上计算[16]、人工智能[17]等多个领域。

特别的，随着人工智能计算需求的不断增大，边缘计算与智能计算相结合的需求越来越迫切[18]，相关的研究也越来越丰富。在期刊数据库Web of Science™核心合集上以边缘计算(检索式：TS=Edge Computing, TS为主题)、边缘计算与人工智能(检索式：TS=Edge Computing AND (TS=Intelligence OR TS=Learning OR TS=Smart))进行检索[19]，对文章数量进行逐年统计，其结果如图1所示。

可以看到，随着主题为边缘计算的文章数量的增加，其中边缘智能计算工作的占比也在不断提高，从2012年的2.23%提高到了2022年的43.56%，已经成为边缘计算研究的焦点。

边缘智能计算将智能计算业务与边缘计算相结合，并将其计算业务分布在整个工作流经过的设备上，可以更好地减少算力浪费、增强隐私保护和加快计算速度。传统的智能计算模型复杂、数据量大、计算量大，单个终端设备保存数据量小、隐私性强。随着终端设备部署智能计算模型需求的增加，如何有效利用云边端架构的优势，对智能计算进行优化加速，成为目前边缘智能计算的研究热点[20]。

目前，对于边缘智能计算的调查研究综述已有很多[21-25]，但从云边端架构的视角出发探讨边缘智能计算业务的综述并不多见。本文聚焦于云边端架构，重点讨论边缘智能计算业务与云边端结合所面临的两个关键问题：计算优化与计算卸载。讨论云边端架构下边缘智能计算优化的应用与研究进展，梳理云边端架构下计算业务卸载策略的研究现状，讨论边缘智能计算业务如何合理地在云边端各部分进行分配。

本文余下的部分结构如下：第2节对云边端架构进行介绍。第3节梳理和讨论云边端架构下边缘智能计算优化的应用和研究。第4节对云边端架构下计算业务计算卸载策略进行了讨论，即计算应该在何时、何地决定在哪里进行。第5节总结了目前云边端架构下的边缘智能计算所面临的挑战与未来可能的研究方向。第6节是结束语。

2 云边端架构

传统云计算模型越来越无法满足计算业务的实时性和安全性，学界和工业界开始讨论在云服务器和终端设备之间部署边缘服务器以平衡算力与延迟之间的矛盾，这种架构称为云边端架构。

在云边端架构中，计算可以发生在云、边、端的每一层，但考虑到网络带宽受限，希望计算尽可

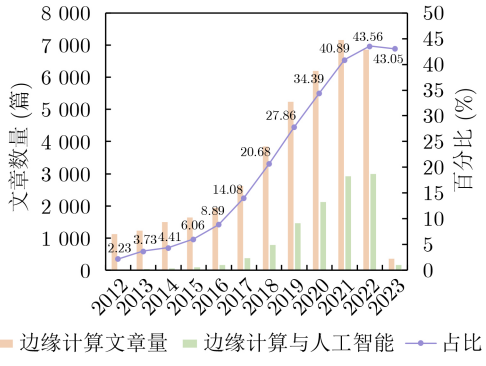

图 1 边缘计算与智能计算结合的趋势

能发生在数据源，即终端设备周围[9]，当数据在终端设备进行计算时，其计算的特点包括：

(1) 计算受限：与云服务器和边缘服务器相比，终端设备的计算能力和效率都十分有限[26]，资源密集型任务的端侧计算可能会带来不能接受的计算时间和能量消耗。

(2) 隐私保护：作为数据的生产者，由于数据不经过网络传输，数据处理过程中隐私可以得到很好的保护[9]。

(3) 网络无关：由于不需要经过网络传输，在终端上进行的计算没有网络延迟，但由于计算能力有限，无网络延迟并不意味着快速。

与终端设备相比，边缘服务器与云服务器的计算时效、计算能效依次增加，然而网络延迟与隐私风险也随之变大。

云端适合资源密集型的业务，即计算量大、数据量小的工作；而终端设备适合处理包含隐私数据或时间敏感型的业务，即运算量小且时效性要求高的工作；边缘节点则介于两者之间。

面对运算量大、实时性强的智能计算业务，如何从算法、硬件、传输等不同维度进行计算优化，使得其可以运行在边缘设备上；以及如何将一个计算业务合理地拆解在云边端的每一个部分，实现计算卸载，实现云边端3层配合，达到总体效果的最优，是边缘智能计算与云边端架构结合所研究的重点[27]。

3　云边端架构下的边缘智能计算优化

当前业内所常用的智能计算业务大致可以分为训练和推理两部分：训练是指通过训练数据生成/调整模型使得模型能够尽量正确地拟合实际数据的分布，目的是得到尽可能准确的模型；而推理是指使用训练好的模型对未知数据进行判断，得到模型的输出。通常来讲训练的计算量远远大于推理。从训练和推理计算发生的位置可以对其进行分类，目前比较流行的云边配合包括：

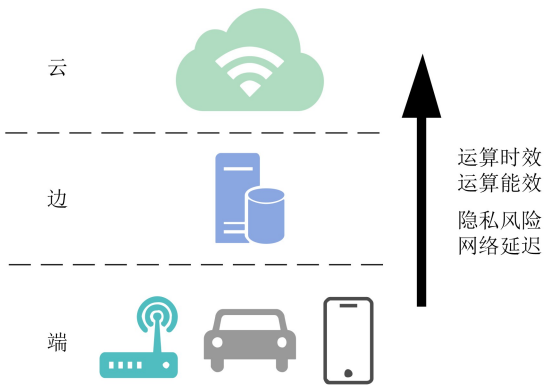

图 2 云边端架构特点

(1) 在云端进行训练，在边缘进行推理。这是目前边缘智能计算框架推荐的应用较多的方案。

(2) 云边端配合训练，在边缘进行推理，共同参与训练过程的每一次迭代，代表算法是联邦学习，适用于有隐私数据参与的场景。

(3) 云边分别进行训练，在边缘进行推理。通过迁移学习等方法，在云端进行第1阶段训练，在边缘端进行第2阶段训练和推理，目的是尽量减少在边缘端训练的计算量。

云边端架构特点如图2所示。

3.1　云端训练，边缘推理

在云端训练，边缘进行推理，主要解决的问题是提高计算速度和减少计算成本。面对算力有限的边缘设备，云端训练出的模型可能无法在边缘设备上直接运行，边缘设备可能也无法运行为云端设计的机器学习平台。因此，为了实现在边缘设备上进行推理，一些为边缘设计的机器学习框架、网络模型轻量化技术被提出。

典型的为边缘计算设计的智能计算编程框架[28]包括TensorFlow Lite, PyTorch Mobile, MindSpore Lite, Paddle Lite等，如表1所示。

考虑到边缘设备有限的存储空间和运算能力，TensorFlow Lite[29]通过剔除不适合在边缘运行的算子来减小框架体积，同时限制模型运算量。然而这降低了其和TensorFlow框架的兼容性。PyTorch Mobile[30]完成了训练模型到部署模型的无缝连接，通过动态量化等方法降低运算量，因此兼容所有PyTorch模型。MindSpore Lite[31]提供模型转化工具，通过模型量化压缩，降低模型的复杂度，使AI模型能在极限环境下部署执行。Paddle Lite[32]是一个终端可部署的轻量化推理引擎，它通过训练完成的模型和设备情况生成轻量化的端侧推理模型，针对移动端设备的机器学习进行优化，压缩模型和二进制文件体积，高效推理，降低内存消耗。

与传统的神经网络模型相比，端侧推理使用的网络模型需要更加轻量化，目前主要使用量化、剪枝、提前退出、轻量化模型等技术减少推理的计算量[33-35]：

量化是指通过低精度变量代替高精度变量。在神经网络的研究中，学者发现降低一些参数变量的精度并不会带来整体模型精度的明显损失，但可以显著压缩模型的大小。文献[36]提出了一种基于激活层的量化方法，使用低比特的整数代替浮点数运算，大幅降低了模型的运算量并且大幅提高了模型在CPU上的运行效率。文献[37]提出了一种训练后量化模型，以避免训练中量化耗时的参数微调，实现了性能接近的低比特量化。

表 1 边缘智能计算框架对比

	TensorFlow Lite	PyTorch Mobile	MindSpore Lite	Paddle Lite
开发者	谷歌	Linux基金会	华为	百度
支持功能	推理	训练+推理	训练+推理	推理
支持的操作系统	Android iOS Linux	iOS Android	Android iOS OpenHarmony Linux Windows	Android iOS Linux Windows macOS

剪枝是指对神经网络的结构进行修改，删除对模型整体不重要的神经元，以此在保证模型精度的条件下减小模型体积。由于不同网络层的权重对网络的贡献不一致，传统的剪枝方法一般分层确定剪枝参数。文献[38]为了在边缘设备上压缩和加速深度神经网络，提出了一种分层的、参数的可变的剪枝方法，遵循贪婪策略对各层参数进行确定，这种方法可以降低神经网络精度剪枝后的下降。在分层剪枝的基础上，文献[39]提出了一种全局的模型剪枝方法，通过量化判断权重改变对网络输出的影响，实现了更细粒度的重要性判断，进一步减小了精度下降。

提前退出与量化和剪枝不同，它会增加模型的参数数量，它通过构建不同的出口，设置阈值，让分类器在结果比较明确时提前做出判断，输出结果，以此来减小推理过程的运算量。针对云边端架构，带有提前退出机制的网络模型一般在终端设备进行一部分计算，通过阈值判断筛选出置信度较低的样本传输至云侧进行进一步计算，置信度较高的样本则直接输出结果，即提前退出。这种模式在通信开销不大的情况下提高模型的准确性且不占用终端设备较多的运算资源。文献[40]通过引入视觉自主自注意力机制(vision transformer)，提出了一种适用于视听数据的提前退出网络，显著降低了运算开销。也有工作[41]关注恶意样本对退出机制的影响，通过对样本的预处理，避免恶意样本使网络不能提前退出造成通信资源的占用。

轻量化模型是指通过改变网络结构以降低推理的运算量，如MobileNet, ShuffleNet等。MobileNet[42]针对卷积神经网络中占用计算资源最多的卷积层进行优化，提出了深度可分离卷积算法，与常规卷积相比，在准确率退化不大的情况下，大大减少了参数量及运算量。ShuffleNet[43]采用逐点分组卷积和通道混洗方法，通过分组卷积降低算力的需求，通过通道混洗可以使各组之间的信息进行交互，保证模型的准确性。

由此可见，云端训练，边缘推理主要关注终端设备算力受限下的计算优化问题，包括架构和算法两个层面，能够实现边缘实时的推理且避免网络延迟，但是训练时需要向云端服务器上传数据，有隐私泄漏的风险。

3.2 云边端配合训练

为了避免隐私的泄露，一些云边端配合进行训练的方法被提出，最有代表性的是联邦学习算法。

联邦学习算法假设训练数据分散在若干个终端设备上，且出于安全性考虑，终端设备不希望其上的数据传输到其他终端或云上而进行训练。它的算法步骤如下[44]：

(1)云端向每一个终端设备下发初始模型。

(2)终端设备使用自己的数据对模型进行训练。

(3)终端设备向云端上传训练好的本地模型及测试精度。

(4)云端对所有终端设备的模型进行聚合(一般是根据终端设备的数据量进行加权平均)，并计算全局模型的精度。如果满足要求，则输出模型，如果不满足则下发聚合后全局模型，回到步骤(2)。

在联邦学习中，云端只接收模型参数和模型精度，因此可以很好地避免终端设备隐私泄露。但是，其算法需要多次迭代，云端与边缘通信次数多，数据量大，受带宽影响很大。为了解决这一问题，基于云边端的3层联邦学习被提出[45]。考虑到降低通信开销会影响模型精度，3层的联邦学习框架提供两次聚合平衡通信与模型精度的矛盾：边缘服务器聚合终端设备的模型，而云端聚合边缘服务器的模型。

由于3层联邦学习可以看作两个双层的联邦学习，故双层联邦学习的结论大部分都可以推广到3层联邦学习，因此3层联邦学习的研究主要集中在两次聚合的配合过程。文献[46]通过分析带参数量化的3层联邦学习的收敛性，进而提出了一种自适应的聚合方案以优化通信策略。文献[47]从安全角度出发，探讨边缘服务器被恶意利用时对云端和终端设备的影响，并据此提出一个安全的3层联邦学习框架，引入数据归一化机制来避免恶意的边缘服务器对学习过程的攻击。

双层联邦学习主要关注资源受限的优化问题，受限的资源包括通信资源、运算资源与数据资源，通过边缘智能计算过程及计算策略的综合优化，可以提高上述资源的利用率以提高模型的准确率并降低运算时间。

通信资源的受限来自联邦学习巨大的通信开销，由于参与的边缘设备数量通常很多，且模型训练需要全体设备多次参与模型迭代，因此通常联邦学习的通信资源都是受限的。将信道与计算过程相结合可以提高数据传输的效率，减小系统的计算时间，文献[48]提出了一种基于概率的调度策略，通过计算边缘计算中联邦学习梯度参数的质量结合每个传输信道情况对数据传输进行优化，最终达到模型的快速收敛以加快计算。

计算受限的优化包括压缩模型、减少训练迭代次数，在减小计算量的同时亦可以减小传输数据量。通过量化、剪枝等方法减小传输模型的尺寸，这与前文所提到的端侧推理中对模型的简化方法类似。引入量化联邦学习的终端设备并不会上传整个模型，而是上传本轮训练所带来的变化量，通过筛选掉变化较小的神经元参数，使其在本轮训练中保持数值不变，可以进一步减少训练的通信开销，如文献[49]提出了一种动态量化的联邦学习方法，可以根据变化量的大小动态地调整量化程度，以降低量化对训练造成的影响。对模型超参进行调整也可以加速模型收敛，如文献[50]证明了在一定条件下调整联邦学习的批处理数量可以加速模型收敛。

数据资源的受限体现在样本数量上，边缘侧的数据通常样本量较小，数据分布有可能不符合独立同分布[51]。联邦学习算法依赖于随机梯度下降，训练数据的独立同分布对于确保随机梯度是全梯度的无偏估计非常重要，因此非独立同分布的训练数据会严重影响训练效果[52]。为解决非独立同分布数据对联邦学习的影响，有学者通过构建统计指标[53]、预训练[54]等方式对边缘数据集进行分组，也有学者通过自注意力机制[55]、客户端加权[56]等方法优化聚合过程。

3.3 云边分别进行训练

为了解决小样本问题及数据分布不均匀的问题，可以在边缘侧进行模型训练和推理。

迁移学习是一种典型的小样本学习方法，它通过在基类上进行学习，再在小样本上进行训练，对参数进行微调，取得了不错的效果。文献[57,58]关注智能摄像头上搭载的深度神经网络的训练过程，提出基于代表性图像的迁移学习及基于单对多、多对单、多对多的智能摄像头迁移学习方法，实现了标注数据量、训练时间的降低和模型准确率的增加。为了解决神经网络在大量物联网终端设备上部署的问题，文献[59]提出了一种模型快速部署和更新方法，通过比较新旧模型的区别来减少传输，并加快训练。

与迁移学习类似的，元学习是一种小样本学习方法，通过对任务空间的学习，来生成与任务无关的模型初始参数，有的方法还会增加记忆模块，能够根据相似的任务进行模型的选择和改进，加快模型收敛的速度、改善服务质量。文献[60]面向智能驾驶，设计了一种基于两阶段元学习的方法，根据数据库中提取的元特征自适应地为车辆在不同场景下的不同计算业务选择合适的机器学习算法。注意到边缘智能计算中样本量小且分布不平均的问题，文献[61]通过在联邦学习中引入元学习思想，提出了一个自适应的部分私有的联邦元学习机制以保证隐私性和收敛性。

迁移学习也可以与联邦学习结合。文献[62]构建了一个迁移联邦学习过程：与传统的联邦学习相比，这一方法不需要一台集中式的云服务器汇总模型，而是通过终端设备互相传递，进一步加强了对隐私的保护。文献[63]结合联邦学习、迁移学习和强化学习，利用联邦学习的本地训练模型来保护隐私，并利用迁移学习通过知识迁移来提高训练效率，最后使用强化学习过滤恶意的终端设备，以实现安全、快速的模型推理。

综上，边缘智能计算的优化主要关注算法在终端、边缘上的加速、分割和卸载。卸载策略主要根据边缘智能计算业务的特点确定。作为边缘计算的核心问题，卸载策略除了根据具体业务的计算卸载研究外，通用的计算卸载策略也是目前研究的热点。

4 云边端架构卸载方案

将计算业务科学合理地分配到云边端每一个层次，称为云边端计算卸载研究，即回答何时进行卸载、由谁决定卸载、卸载到哪里三个问题。对于不同的业务的优化约束条件和不同场景的网络传输条件，计算任务的卸载策略不尽相同。常用的约束条件包括任务时延和系统能耗[64-67]，也有研究同时考虑服务质量[68, 69]、隐私保护程度[70]等。以下主要针对时延和能耗两个研究热点进行综述。

值得一提的是，由于边缘计算包含大量时间敏感型业务，因此单纯考虑能耗的卸载方法可能会造成不能接受的时延，因此大部分以能耗为指标的卸载研究也会同时考虑时间指标。

4.1 针对任务时延优化的卸载方案

任务时延是云边端体系下服务质量的重要指标。

任务时延通常可以分为3部分[71, 72]：网络传输时延、传输后在队列中等待时延、计算时延。

云边端系统的各部分的通信成本和计算能力各有不同，边缘侧通信速度快，但计算速度慢；而云侧计算时间短，通信时间长。因此需要根据不同情景平衡通信和计算带来的时间开销。

马勇等人[73]提出了一种云边端架构下中心式的调度模式，如图3所示。云服务器获取终端设备的位置信息，并基于深度学习进行预测，得到终端设备预测轨迹。根据历史轨迹和预测轨迹，计算设备的综合距离，将设备分为热点区域设备和离散设备。对热点区域设备，云服务器计算热点区域与边缘服务器匹配度，根据匹配度得到热点区域计算簇(边缘服务器的集合)，热点区域设备向计算簇控制节点(由距离热点区域最近的边缘服务器担任)发起卸载，由控制节点将计算任务分发至边缘服务器执行。对离散设备，云服务器计算设备与边缘服务器的匹配度，设备根据匹配度直接向边缘服务器发起卸载，如图4所示。

该方法的优点是终端可以选择最优的边缘服务器进行运算，但终端与云服务器的通信通常是受限的。该方法需要终端与云端多次通信，在网络延迟较大甚至中断时无法有效进行计算卸载。

Ren等人[74]证明了在终端设备使用时分多址无线网络传输，且信道服从独立同分布的瑞利衰落时，计算资源的分配存在最优解。该文章假设终端只产生数据和接收计算结果，所有计算任务全部卸载到边缘服务器及云服务器。终端首先上传所有数据到边缘服务器，边缘服务器做出判断，划分计算量，并上传云服务器需要进行运算的数据，然后边缘服务器和云服务器对数据进行并行的计算。作者证明了这一过程的最优化问题是一个严格凸函数，由此确定了理论最优卸载策略。

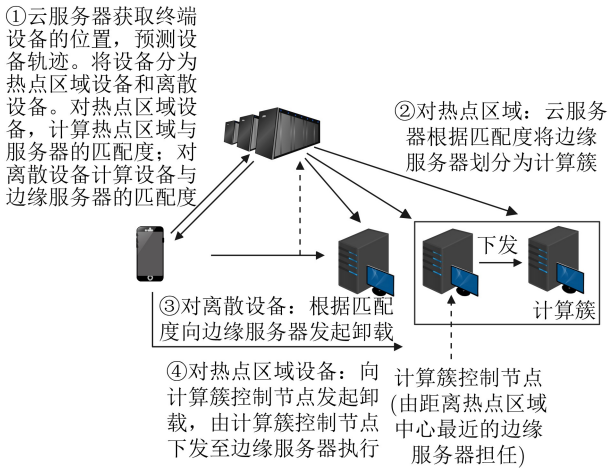

图 3 中心式计算卸载流程

该方法的计算划分发生在边缘服务器上，因此在与云端网络通信中断时也可使用，并且是理论最优策略。然而作者忽略了计算任务划分的粒度问题及依赖关系，假设任务是完全可分的且没有相互依赖。同时该结论只适用于终端设备使用时分多址技术进行无线网络传输这一过程，终端与边缘服务器的关系固定，无法根据网络状态、资源占用情况等进行调整。假设计算全部发生在边缘服务器及云服务，降低了优化问题的维度，但浪费了终端设备的低时延计算能力。

为了克服云服务器统一调度造成的巨大延迟和固定边缘服务器造成的算力损耗，Lin等人[75]提出了由边缘服务器主导的局部调度算法：引入了一种称为“两种选择的力量”的简单策略，实现了基于边缘服务器的轻量化负载均衡，如图5所示。

首先终端和边缘通过云服务器和广播发现彼此，之后每个终端选择一个延迟最小的边缘服务器作为其“守护节点”。需要执行卸载时，首先将任务信息传输到守护节点，若守护节点有空闲资源，则将数据传输到守护节点进行计算。若没有空闲资源，则守护节点任意选择两个其他边缘节点，选择其中负载小的作为执行节点，重定向数值发送至执行节点，所谓负载小，是指预期的运行时间小。同时，该调度系统考虑了时间敏感性任务的有限性，这个简单的策略被证明是有效的。

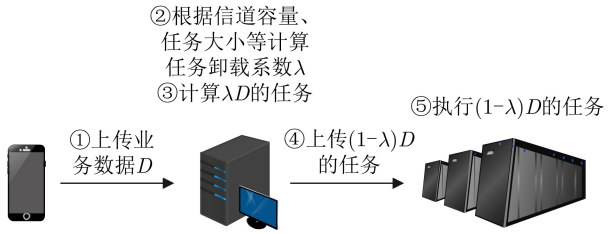

图 4 边缘式计算卸载流程

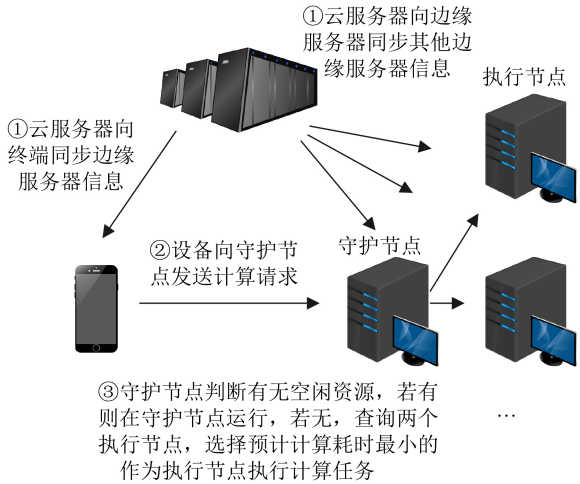

图 5 基于“两种选择的力量”策略的卸载流程

该方法的优势在于由边缘服务器进行信息不全的快速卸载策略制定，由此平衡了卸载策略生成速度和卸载策略实施效果，但云服务器作为系统中运算效率最高的计算平台并没有有效参与计算。

4.2 综合时延和系统能耗优化的卸载方案

移动终端设备的电池容量通常是有限的，将计算卸载到边缘和云可以有效提高移动终端设备的续航能力。边缘计算服务通常对时间较为敏感，单纯考虑能耗可能会带来不能接受的任务时延。因此研究的热点为结合时延指标对卸载进行约束，在满足时延要求的条件下实现系统耗能最小，能耗主要考虑计算能耗和信息传输能耗。

一般而言，能耗的计算公式为[76]

$$\text{计算能耗} = \text{设备功率}\times\text{运行时间} \tag{1}$$

$$\text{运行时间} = \text{所需CPU周期数}\times\text{CPU周期时长} \tag{2}$$

$$\text{传输能耗} = (\text{传输功率} + \text{接收功率})\times\text{传输时间} \tag{3}$$

$$\text{传输时间} = \text{传输数据量}/\text{传输效率} \tag{4}$$

由于云边端各层次设备功率、运算效率、传输效率不一致，因此综合时延和能耗的卸载方案也可以被视为一个为有权重的时延优化方案。

徐佳等人[77]考虑了多任务场景下的计算卸载，讨论了一个存在依赖关系的多任务卸载体系，同时考虑等待时间的能量消耗，提出了一个基于粒子群优化算法的多任务云边端卸载策略。该策略的核心是一个分段函数，在满足时间要求时，分段函数仅与能耗有关；不满足时间要求时，分段函数与时间和功耗同时有关。通过迭代求得该分段函数的最小值进行计算卸载。仿真结果显示，该方法在能耗和计算时延上都有不错的表现。但由于时延与能耗是相互存在相互关联的，因此该方法的计算复杂度较高。

在引入任务依赖的基础上，李强等人[78]进一步考虑了时变信道下边缘计算任务的卸载。采用A3C深度强化学习方法，将计算卸载问题转化为一个马尔可夫决策过程。使用多线程智能体，首先在离线环境中使用模拟数据进行训练；在实际应用环境中，根据真实数据进行多线程的推理和异步训练，适应真实条件。与基类方法相比，时延和系统能耗得到了很好的控制。但是该算法只考虑了单一设备至单一边缘服务器，没有考虑多信道波动下边缘服务器的选择和用户竞争。

彭昇等人[79]考虑了一个多终端多边缘服务器的云边端计算场景：终端移动设备通过无线网与基站进行连接，基站之间通过中央基站相连。算法考虑设备用户的移动轨迹，依据路径选择最优的两个基站进行计算卸载。在同一个时刻，一个基站负责此时刻的计算，另一个基站负责接收下一时刻计算的数据。该方法有效地解决了设备移动带来的网络通信条件的变化，并实现了较好的系统功耗控制。但是对于时延仅做了定性的判断，还有优化的空间。

5 云边端架构下边缘智能计算挑战与研究趋势

通过对云边端架构下边缘智能计算优化及计算卸载策略的梳理可以看出，尽管边缘智能计算在学术界和工业界已经有了广泛的研究和应用，该领域仍然有着巨大的研究潜力和挑战，下面总结云边端架构下边缘智能计算的未来发展方向及挑战，希望以下这些讨论能够促进云边端架构与边缘智能计算的融合、发展和创新。

5.1 云边端架构与具体计算业务的深度融合

由于不同计算业务的可分割粒度、子任务间的依赖关系、计算量数据量之比、任务数据的隐私程度都有所不同，因此与单纯面向架构的计算卸载、运算相比，结合具体业务后能够进一步优化云边端架构的消耗，激发云边端架构的潜能。

边缘智能计算的训练过程是典型的资源密集型业务，即运算量大且时效性要求不高的计算；而推理过程则是时间敏感型业务，运算量较小但要求较高的时效性。针对它们的特点就可以更合理地进行计算业务划分。同时，由于端侧的计算资源受限，针对终端的计算优化也是未来云边端架构的解决方案之一，而架构与具体业务的结合是这一优化的前提。

以作者参与的重点研发计划子任务为例，目的是构建一个由终端监测设备(传感器为红外和可见光相机)、地面处理站、指挥中心构成的森林火灾监测系统。由于带宽受限，监测设备无法将获取的全部数据上传至地面处理站，因此需要在监测设备上进行初步运算，对数据进行筛选。此时监测设备需要处理的数据量为3层中最大，但运算能力是3层中的最低，数据量与运算能力之间产生了极大的矛盾。考虑到带宽有限，将部分火点识别的算法卸载到检测设备上进行。可见光传感器分辨率高，火点识别运算量大，准确率高；红外数据分辨率低，火点识别运算量小，虚警率高；考虑到传感器特点和业务流程，由检测设备进行基于红外图像的火点初步识别，并将疑似火点的图像传输至地面处理站，由处理站进行基于可见光图像的火点精细识别，剔除虚警后，生成观测需求下传至监测设备，并将火点图像及信息上传至指挥中心，以此平衡传输与处理之间的矛盾。

5.2 终端设备差异的解决方案

终端设备数量巨大带来的问题之一是运算能

力、数据结构、数据分布、通信状况都有所不同。首先，运算能力的差异可能使得算法难以部署，如某些单片机可能无法运行某些神经网络模型或时延不可接受。不同终端设备的兼容性也可能导致算法无法运行，目前各个边缘智能计算框架标准和接口尚不统一。

其次是数据的差异，数据的差异首先是数据结构的差异，这使得不同终端的数据难以被同样的程序处理；除了数据结构，数据的分布也可能有差异，数据的独立同分布是神经网络准确的前提之一，如果数据分布差异过大，则需要对数据进行进一步的细分。

最后是通信状况的差异，这种差异带主要影响计算的卸载策略，目前的研究目标多数是系统层面的最优，然而这种优化边界条件可能导致通信状况较差的设备被忽视，从而承受极差的服务质量，由此可能带来用户体验与公平性的问题。

5.3 多实体参与的边缘智能计算的激励及可信计算

在边缘智能计算特别是联邦学习中，计算优化与计算卸载过程可能涉及不同利益组织的实体，这些实体互相配合需要有效的激励机制进行支撑。考虑到边缘环境松散及去中心化的特点，目前学术界通过引入区块链等技术，进行了一些激励方法研究，但这种引入势必会挤占边缘计算场景中宝贵的传输、运算和储存资源。边缘计算场景呼唤更简单、更公平且有效的激励方案。

进一步的，多实体参与的边缘智能计算可能出现恶意参与者，窃取数据或者影响计算可靠性。如何识别、记录、排除恶意参与的实体，不影响系统开放性的同时尽可能提高系统的安全性，实现可信的边缘智能计算，也是目前边缘智能计算的挑战之一。

5.4 通信受限条件下的边缘计算智能卸载

计算卸载策略是云边端架构的核心算法，卸载模式与通信环境有着密切的关系，当与云服务器的通信时延较长时，基于边缘服务器的决策因为反应迅速所以效果较好；反之，与云服务器通信时延短时，则由云服务器进行总体统筹效果最佳。

目前的研究虽然大都考虑了通信对计算卸载的影响，但策略相对固定，很少有根据通信环境而对卸载策略进行改进的方法。应进一步发挥节点的自组织性，实现通信受限场景下策略的智能选择，提高系统的鲁棒性。

6 结束语

本文结合云边端架构与边缘智能计算，从架构的视角出发，回顾了云边端架构下边缘智能计算的两大关键问题：计算优化与计算卸载。分析了云边端架构下边缘智能计算及计算卸载的研究方向与研究进展。最后总结了云边端架构下边缘智能计算的几个挑战和未来的研究趋势。本文希望这些对云边端架构边缘智能计算的总结和思考，能够促进边缘计算与人工智能的进一步结合，为这一领域带来进步和发展。

参 考 文 献

[1] 工业和信息化部. 2022年通信业统计公报[EB/OL]. https://www.gov.cn/xinwen/2023-02/02/content_5739680.htm, 2023.
Ministry of Industry and Information Technology of the People's Republic of China. 2022 Communication industry Statistical Bulletin[EB/OL]. https://www.gov.cn/xinwen/2023-02/02/content_5739680.htm, 2023.

[2] 刘俊旭, 孟小峰. 机器学习的隐私保护研究综述[J]. 计算机研究与发展, 2020, 57(2): 346–362. doi: 10.7544/issn1000-1239.2020.20190455.
LIU Junxu and MENG Xiaofeng. Survey on privacy-preserving machine learning[J]. *Journal of Computer Research and Development*, 2020, 57(2): 346–362. doi: 10.7544/issn1000-1239.2020.20190455.

[3] XING Tong, BARBALACE A, OLIVIER P, *et al*. H-container: Enabling heterogeneous-ISA container migration in edge computing[J]. *ACM Transactions on Computer Systems*, 2021, 39(1/4): 5. doi: 10.1145/3524452.

[4] CHEN Ying, ZHAO Fengjun, CHEN Xin, *et al*. Efficient multi-vehicle task offloading for mobile edge computing in 6G networks[J]. *IEEE Transactions on Vehicular Technology*, 2022, 71(5): 4584–4595. doi: 10.1109/TVT.2021.3133586.

[5] LIU Chunyu, WANG Kailin, ZHANG Heli, *et al*. Rendered tile reuse scheme based on FoV prediction for MEC-assisted wireless VR service[J]. *IEEE Transactions on Network Science and Engineering*, 2023, 10(3): 1709–1721. doi: 10.1109/TNSE.2023.3234029.

[6] HAZRA A, RANA P, ADHIKARI M, *et al*. Fog computing for next-generation internet of things: Fundamental, state-of-the-art and research challenges[J]. *Computer Science Review*, 2023, 48: 100549. doi: 10.1016/j.cosrev.2023.100549.

[7] BHARANY S, SHARMA S, KHALAF O I, *et al*. A systematic survey on energy-efficient techniques in sustainable cloud computing[J]. *Sustainability*, 2022, 14(10): 6256. doi: 10.3390/su14106256.

[8] DAMSGAARD H J, OMETOV A, and NURMI J. Approximation opportunities in edge computing hardware: A systematic literature review[J]. *ACM Computing Surveys*, 2023, 55(12): 252. doi: 10.1145/3572772.

[9] SHI Weisong, CAO Jie, ZHANG Quan, *et al*. Edge computing: Vision and challenges[J]. *IEEE Internet of Things Journal*, 2016, 3(5): 637–646. doi: 10.1109/JIOT.2016.2579198.

[10] 朱特浩. 边缘计算在军事信息系统智能化发展中的应用[J]. 火力与指挥控制, 2021, 46(8): 5–11. doi: 10.3969/j.issn.1002-0640.2021.08.002.
ZHU Tehao. Analysis of the application of edge computing in the intelligent development of military information system[J]. *Fire Control & Command Control*, 2021, 46(8): 5–11. doi: 10.3969/j.issn.1002-0640.2021.08.002.

[11] CONTRERAS L M, SOLANO A, CANO F, *et al*. Analysis of network function sharing in content delivery network-as-a-service slicing scenarios[J]. *International Journal of Network Management*, 2023, 33(4): e2221. doi: 10.1002/nem.2221.

[12] KHAN L U, YAQOOB I, TRAN N H, *et al*. Edge-computing-enabled smart cities: A comprehensive survey[J]. *IEEE Internet of Things Journal*, 2020, 7(10): 10200–10232. doi: 10.1109/JIOT.2020.2987070.

[13] LIN Jie, YANG Peng, ZHANG Ning, *et al*. Low-latency edge video analytics for on-road perception of autonomous ground vehicles[J]. *IEEE Transactions on Industrial Informatics*, 2023, 19(2): 1512–1523. doi: 10.1109/TII.2022.3181986.

[14] 刘子杰, 王凯, 王亚刚, 等. 工业互联网端边云协同数据同步方案设计与实现[J]. 计算机应用研究, 2022, 39(3): 821–825. doi: 10.19734/j.issn.1001-3695.2021.09.0349.
LIU Zijie, WANG Kai, WANG Yagang, *et al*. Design and implementation of end-to-end cloud collaborative data synchronization scheme for industrial Internet[J]. *Application Research of Computers*, 2022, 39(3): 821–825. doi: 10.19734/j.issn.1001-3695.2021.09.0349.

[15] SHARMA V, TRIPATHI A K, and MITTAL H. Technological revolutions in smart farming: Current trends, challenges & future directions[J]. *Computers and Electronics in Agriculture*, 2022, 201: 107217. doi: 10.1016/j.compag.2022.107217.

[16] 周锦雯, 刘乃金, 陈清霞. 基于分布式深度学习的多星计算卸载策略[J]. 中国空间科学技术, 2023, 43(2): 73–80. doi: 10.16708/j.cnki.1000-758X.2023.0022.
ZHOU Jinwen, LIU Naijin, and CHEN Qingxia. Multi-satellite task offloading method based on distributed deep learning[J]. *Chinese Space Science and Technology*, 2023, 43(2): 73–80. doi: 10.16708/j.cnki.1000-758X.2023.0022.

[17] SUN Cjuan, LI Xiuhua, WEN Junhao, *et al*. Federated deep reinforcement learning for recommendation-enabled edge caching in mobile edge-cloud computing networks[J]. *IEEE Journal on Selected Areas in Communications*, 2023, 41(3): 690–705. doi: 10.1109/JSAC.2023.3235443.

[18] DENG Shuiguang, ZHAO Hailiang, FANG Weijia, *et al*. Edge intelligence: The confluence of edge computing and artificial intelligence[J]. *IEEE Internet of Things Journal*, 2020, 7(8): 7457–7469. doi: 10.1109/JIOT.2020.2984887.

[19] Clarivate. Document search - web of science core collection[EB/OL]. https://www.webofscience.com/wos/woscc/basic-search, 2023.

[20] CHEN Jiasi and RAN Xukan. Deep learning with edge computing: A review[J]. *Proceedings of the IEEE*, 2019, 107(8): 1655–1674. doi: 10.1109/JPROC.2019.2921977.

[21] ZHANG Jing and TAO Dacheng. Empowering things with intelligence: A survey of the progress, challenges, and opportunities in artificial intelligence of things[J]. *IEEE Internet of Things Journal*, 2021, 8(10): 7789–7817. doi: 10.1109/JIOT.2020.3039359.

[22] 张晓东, 张朝昆, 赵继军. 边缘智能研究进展[J/OL]. 计算机研究与发展: 1–22. http://kns.cnki.net/kcms/detail/11.1777.tp.20230310.1731.006.html, 2023.
ZHANG Xiaodong, ZHANG Chaokun, and ZHAO Jijun. State-of-the-art survey on edge intelligence[J/OL]. *Journal of Computer Research and Development*: 1–22. http://kns.cnki.net/kcms/detail/11.1777.tp.20230310.1731.006.html, 2023.

[23] ZHOU Zhi, CHEN Xu, LI En, *et al*. Edge intelligence: Paving the last mile of artificial intelligence with edge computing[J]. *Proceedings of the IEEE*, 2019, 107(8): 1738–1762. doi: 10.1109/JPROC.2019.2918951.

[24] HU Haizhou and JIANG Congfeng. Edge intelligence: Challenges and opportunities[C]. 2020 International Conference on Computer, Information and Telecommunication Systems (CITS), Hangzhou, China, 2020: 1–5. doi: 10.1109/CITS49457.2020.9232575.

[25] XU Dianlei, LI Tong, LI Yong, *et al*. Edge intelligence: Empowering intelligence to the edge of network[J]. *Proceedings of the IEEE*, 2021, 109(11): 1778–1837. doi: 10.1109/JPROC.2021.3119950.

[26] KHAN W Z, AHMED E, HAKAK S, *et al*. Edge computing: A survey[J]. *Future Generation Computer Systems*, 2019, 97: 219–235. doi: 10.1016/j.future.2019.02.050.

[27] 刘通, 方璐, 高洪皓. 边缘计算中任务卸载研究综述[J]. 计算机科学, 2021, 48(1): 11–15. doi: 10.11896/jsjkx.200900217.
LIU Tong, FANG Lu, and GAO Honghao. Survey of task offloading in edge computing[J]. *Computer Science*, 2021, 48(1): 11–15. doi: 10.11896/jsjkx.200900217.

[28] 中国信通院. AI框架发展白皮书[M]. 北京: 中国信通院, 2022: 35–36.
China Academy of Information and Communications Technology. White Paper on AI Framework Development[M]. Beijing: China Academy of Information

and Communications Technology, 2022: 35–36.

[29] TensorFlow. TensorFlow lite[EB/OL]. https://tensorflow.google.cn/lite/guide?hl=zh-cn, 2021.

[30] The Linux Foundation. PYTORCH MOBILE[EB/OL]. https://pytorch.org/mobile/home/, 2023.

[31] MindSpore. MindSpore lite[EB/OL]. https://www.mindspore.cn/lite?version=/master/, 2023.

[32] PaddlePaddle Developers. Paddle lite - 端侧轻量化推理引擎[EB/OL]. https://paddlelite.paddlepaddle.org.cn/, 2023.
PaddlePaddle Developers. Paddle lite-End-side lightweight inference engine[EB/OL]. https://paddlelite.paddlepaddle.org.cn/, 2023

[33] 李博闻. 深度神经网络量化及其硬件加速研究[D]. [博士论文], 浙江大学, 2022. doi: 10.27461/d.cnki.gzjdx.2022.000973.
LI Bowen. Quantization and hardware acceleration for deep neural network[D]. [Ph. D. dissertation], Zhejiang University, 2022. doi: 10.27461/d.cnki.gzjdx.2022.000973.

[34] 林景栋, 吴欣怡, 柴毅, 等. 卷积神经网络结构优化综述[J]. 自动化学报, 2020, 46(1): 24–37. doi: 10.16383/j.aas.c180275.
LIN Jingdong, WU Xinyi, CHAI Yi, *et al*. Structure optimization of convolutional neural networks: A survey[J]. *Acta Automatica Sinica*, 2020, 46(1): 24–37. doi: 10.16383/j.aas.c180275.

[35] ZOUMPEKAS T, SALAMÓ M, and PUIG A. effective early stopping of point cloud neural networks[C]. The 19th International Conference on Modeling Decisions for Artificial Intelligence (MDAI), Sant Cugat, Spain, 2022: 156–167. doi: 10.1007/978-3-031-13448-7_13.

[36] MEI Shaohui, CHEN Xiaofeng, ZHANG Yifan, *et al*. Accelerating convolutional neural network-based hyperspectral image classification by step activation quantization[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2022, 60: 5502012. doi: 10.1109/TGRS.2021.3058321.

[37] WANG Peisong, CHEN Weihan, HE Xianyu, *et al*. Optimization-based post-training quantization with bit-split and stitching[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023, 45(2): 2119–2135. doi: 10.1109/TPAMI.2022.3159369.

[38] LI Guangli, MA Xiu, WANG Xueying, *et al*. Optimizing deep neural networks on intelligent edge accelerators via flexible-rate filter pruning[J]. *Journal of Systems Architecture*, 2022, 124: 102431. doi: 10.1016/j.sysarc.2022.102431.

[39] YU Fang, CUI Li, WANG Pengcheng, *et al*. EasiEdge: A novel global deep neural networks pruning method for efficient edge computing[J]. *IEEE Internet of Things Journal*, 2021, 8(3): 1259–1271. doi: 10.1109/JIOT.2020.3034925.

[40] BAKHTIARNIA A, ZHANG Qi, and IOSIFIDIS A. Single-layer vision transformers for more accurate early exits with less overhead?[J]. *Neural Networks*, 2022, 153: 461–473. doi: 10.1016/j.neunet.2022.06.038.

[41] QIU Han, ZHANG Tianwei, ZHANG Tianzhu, *et al*. DefQ: Defensive quantization against inference slow-down attack for edge computing[J]. *IEEE Internet of Things Journal*, 2023, 10(4): 3243–3251. doi: 10.1109/JIOT.2021.3138935.

[42] HOWARD A G, ZHU Menglong, CHEN Bo, *et al*. MobileNets: Efficient convolutional neural networks for mobile vision applications[EB/OL]. https://arxiv.org/abs/1704.04861, 2017.

[43] ZHANG Xiangyu, ZHOU Xinyu, LIN Mengxiao, *et al*. ShuffleNet: An extremely efficient convolutional neural network for mobile devices[C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, USA, 2018: 6848–6856. doi: 10.1109/CVPR.2018.00716.

[44] 王健宗, 孔令炜, 黄章成, 等. 联邦学习算法综述[J]. 大数据, 2020, 6(6): 64–82. doi: 10.11959/j.issn.2096-0271.2020055.
WANG Jianzong, KONG Lingwei, HUANG Zhangcheng, *et al*. Research review of federated learning algorithms[J]. *Big Data Research*, 2020, 6(6): 64–82. doi: 10.11959/j.issn.2096-0271.2020055.

[45] LIU Lumin, CHANG Jun, SONG S H, *et al*. Client-edge-cloud hierarchical federated learning[C]. 2020 IEEE International Conference on Communications, Dublin, Ireland 2020: 1–6. doi: 10.1109/ICC40277.2020.9148862.

[46] LIU Lumin, ZHANG Jun, SONG Shenghui, *et al*. Hierarchical federated learning with quantization: Convergence analysis and system design[J]. *IEEE Transactions on Wireless Communications*, 2023, 22(1): 2–18. doi: 10.1109/TWC.2022.3190512.

[47] ZHOU Hongliang, ZHENG Yifeng, HUANG Hejiao, *et al*. Toward robust hierarchical federated learning in internet of vehicles[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 24(5): 5600–5614. doi: 10.1109/TITS.2023.3243003.

[48] REN Jinke, HE Yinghui, WEN Dingzhu, *et al*. Scheduling for cellular federated edge learning with importance and channel awareness[J]. *IEEE Transactions on Wireless Communications*, 2020, 19(11): 7690–7703. doi: 10.1109/TWC.2020.3015671.

[49] FENG Wenjun and ZHANG Xian. Wireless federated learning with dynamic quantization and bandwidth adaptation[J]. *IEEE Wireless Communications Letters*, 2022, 11(11): 2335–2339. doi: 10.1109/LWC.2022.3202645.

[50] REN Jinke, YU Guanding, and DING Guangyao. Accelerating DNN training in wireless federated edge learning systems[J]. *IEEE Journal on Selected Areas in Communications*, 2021, 39(1): 219–232. doi: 10.1109/JSAC.

2020.3036971.

[51] ZHU Hangyu, XU Jinjin, LIU Shiqing, *et al*. Federated learning on non-IID data: A survey[J]. *Neurocomputing*, 2021, 465: 371–390. doi: 10.1016/j.neucom.2021.07.098.

[52] ZHAO Yue, LI Meng, LAI Liangzhen, *et al*. Federated learning with non-IID data[EB/OL]. https://arxiv.org/abs/1806.00582, 2022.

[53] CHEN Aiguo, FU Yang, WANG Lingfu, *et al*. DWFed: A statistical-heterogeneity-based dynamic weighted model aggregation algorithm for federated learning[J]. *Frontiers in Neurorobotics*, 2022, 16: 1041553. doi: 10.3389/fnbot.2022.1041553.

[54] JAMALI-RAD H, ABDIZADEH M, and SINGH A. Federated learning with taskonomy for non-IID data[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2022: 1–12. doi: 10.1109/TNNLS.2022.3152581..

[55] XU Yawen, LI Xiaojun, YANG Zeyu, *et al*. Robust communication strategy for federated learning by incorporating self-attention[C]. SPIE 11584, 2020 International Conference on Image, Video Processing and Artificial Intelligence, Shanghai, China, 2020: 115841F. doi: 10.1117/12.2581491.

[56] ARAFEH M, OULD-SLIMANE H, OTROK H, *et al*. Data independent warmup scheme for non-IID federated learning[J]. *Information Sciences*, 2023, 623: 342–360. doi: 10.1016/j.ins.2022.12.045.

[57] LU C H and LIN Xiaozong. Toward direct edge-to-edge transfer learning for IoT-enabled edge cameras[J]. *IEEE Internet of Things Journal*, 2021, 8(6): 4931–4943. doi: 10.1109/JIOT.2020.3034153.

[58] LU C H and ZHOU Yangming. Direct edge-to-edge many-to-many latent feature transfer learning[J]. *IEEE Internet of Things Journal*, 2022, 9(12): 10048–10060. doi: 10.1109/JIOT.2021.3117991.

[59] HSU T H, WANG Zhihao, and SEE A R. A Cloud-edge-smart IoT architecture for speeding up the deployment of neural network models with transfer learning techniques[J]. *Electronics*, 2022, 11(14): 2255. doi: 10.3390/electronics11142255.

[60] CHEN Dawei, LIU Yinchen, KIM B, *et al*. Edge computing resources reservation in vehicular networks: A meta-learning approach[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(5): 5634–5646. doi: 10.1109/TVT.2020.2983445.

[61] DONG Fang, GE Xinghua, LI Qinya, *et al*. PADP-FedMeta: A personalized and adaptive differentially private federated meta learning mechanism for AIoT[J]. *Journal of Systems Architecture*, 2023, 134: 102754. doi: 10.1016/j.sysarc.2022.102754.

[62] LIU Yi, PENG Jialiang, KANG Jiawen, *et al*. A secure federated learning framework for 5G networks[J]. *IEEE Wireless Communications*, 2020, 27(4): 24–31. doi: 10.1109/MWC.01.1900525.

[63] CHENG Yanyu, LU Jianyuan, NIYATO D, *et al*. Federated transfer learning with client selection for intrusion detection in mobile edge computing[J]. *IEEE Communications Letters*, 2022, 26(3): 552–556. doi: 10.1109/LCOMM.2022.3140273.

[64] 李强, 杜婷婷, 童钊, 等. 移动边缘计算中基于深度强化学习的依赖任务卸载研究[J/OL]. 小型微型计算机系统: 1–8. https://doi.org/10.20009/j.cnki.21-1106/TP.2021-0823, 2023.

LI Qiang, DU Tingting, TONG Zhao, *et al*. Dependent task offload based on deep reinforcement learning in mobile edge computing[J/OL]. *Journal of Chinese Computer Systems:* 1–8. https://doi.org/10.20009/j.cnki.21-1106/TP.2021-0823, 2023.

[65] WU Chao, ZHANG Yaoxue, and DENG Yongheng. Toward fast and distributed computation migration system for edge computing in IoT[J]. *IEEE Internet of Things Journal*, 2019, 6(6): 10041–10052. doi: 10.1109/JIOT.2019.2935120.

[66] TANG Ming and WONG V W S. Deep reinforcement learning for task offloading in mobile edge computing systems[J]. *IEEE Transactions on Mobile Computing*, 2022, 21(6): 1985–1997. doi: 10.1109/TMC.2020.3036871.

[67] BABAR M and KHAN M S. ScalEdge: A framework for scalable edge computing in Internet of things-based smart systems[J]. *International Journal of Distributed Sensor Networks*, 2021, 17(7). doi: 10.1177/15501477211035332.

[68] HAN Dongsheng, LIU Yu, and NI Junhong. Research on multinode collaborative computing offloading algorithm based on minimization of energy consumption[J]. *Wireless Communications and Mobile Computing*, 2020, 2020: 8858298. doi: 10.1155/2020/8858298.

[69] 胡世红. 边缘计算中资源动态调度的QoS优化技术研究[D]. [博士论文], 江南大学, 2021. doi: 10.27169/d.cnki.gwqgu.2021.001939.

HU Shihong. Research on QoS optimization technologies of dynamic resource scheduling in edge computing[D]. [Ph. D. dissertation], Jiangnan University, 2021. doi: 10.27169/d.cnki.gwqgu.2021.001939.

[70] 张成虎, 李鹏旭, 王琪. 网络金融犯罪预警系统研究——基于区块链和边缘计算[J]. 情报杂志, 2023, 42(1): 59–65. doi: 10.3969/j.issn.1002-1965.2023.01.009.

ZHANG Chenghu, LI Pengxu, and WANG Qi. Research on network financial crime early warning system——based on blockchain and edge computing[J]. *Journal of Intelligence*, 2023, 42(1): 59–65. doi: 10.3969/j.issn.1002-1965.2023.01.009.

[71] 张俊娜, 鲍想, 陈家伟, 等. 一种联合时延和能耗的依赖性任务卸载方法[J/OL]. 计算机研究与发展, 1–13. http://kns.cnki.

net/kcms/detail/11.1777.TP.20230213.0839.002.html, 2023.
ZHANG Junna, BAO Xiang, CHENG Jiawei, *et al*. A dependent task offloading method for joint delay and energy consumption[J/OL]. *Journal of Computer Research and Development*, 1–13. http://kns.cnki.net/kcms/detail/11.1777.TP.20230213.0839.002.html, 2023.

[72] DING Xinhui and ZHANG Wenjuan. Computing unloading strategy of massive internet of things devices based on game theory in mobile edge computing[J]. *Mathematical Problems in Engineering*, 2021, 2021: 2163965. doi: 10.1155/2021/2163965.

[73] 马勇, 戴梦轩, 夏云霓, 等. 一种基于人群分类的边缘计算任务卸载方法[P]. 中国, 115878227A, 2023.
MA Yong, DAI Mengxuan, XIA Yunni, *et al*. A method of edge computing task unloading based on crowd classification[P]. CN, 115878227A, 2023.

[74] REN Jinke, YU Guanding, HE Yinghui, *et al*. Collaborative cloud and edge computing for latency minimization[J]. *IEEE Transactions on Vehicular Technology*, 2019, 68(5): 5031–5044. doi: 10.1109/TVT.2019.2904244.

[75] LIN Li, LI Peng, XIONG Jinbo, *et al*. Distributed and application-aware task scheduling in edge-clouds[C]. 2018 14th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN), Shenyang, China, 2018: 165–170. doi: 10.1109/MSN.2018.000-1.

[76] REN Jinke, YU Guanding, CAI Yunlong, *et al*. Latency optimization for resource allocation in mobile-edge computation offloading[J]. *IEEE Transactions on Wireless Communications*, 2018, 17(8): 5506–5519. doi: 10.1109/TWC.2018.2845360.

[77] 徐佳, 李学俊, 丁瑞苗, 等. 移动边缘计算中能耗优化的多重资源计算卸载策略[J]. 计算机集成制造系统, 2019, 25(4): 954–961. doi: 10.13196/j.cims.2019.04.018.
XU Jia, LI Xuejun, DING Ruimiao, *et al*. Energy efficient multi-resource computation offloading strategy in mobile edge computing[J]. *Computer Integrated Manufacturing Systems*, 2019, 25(4): 954–961. doi: 10.13196/j.cims.2019.04.018.

[78] 李强, 仪晋辉, 杜婷婷, 等. 移动边缘计算中基于A3C的依赖任务卸载与资源分配[J]. 计算机工程, 2023, 49(6): 42–52. doi: 10.19678/j.issn.1000-3428.0066095.
LI Qiang, YI Jinhui, DU Tingting, *et al*. Dependent task offloading and resource allocation based on A3C in mobile edge computing[J]. *Computer Engineering*, 2023, 49(6): 42–52. doi: 10.19678/j.issn.1000-3428.0066095.

[79] 彭昇, 赵建保, 魏敏捷, 等. 基于移动边缘计算的任务卸载优化[J]. 计算机系统应用, 2023, 32(4): 262–267. doi: 10.15888/j.cnki.csa.009013.
PENG Sheng, ZHAO Jianbao, WEI Minjie, *et al*. Task offload optimization based on mobile edge computing[J]. *Computer Systems & Applications*, 2023, 32(4): 262–267. doi: 10.15888/j.cnki.csa.009013.

董裕民：男，博士生，研究方向为边缘计算、知识工程等.
张 静：女，硕士，高级工程师，研究方向为遥感信息质量控制、分布式智能决策等.
谢昌佐：男，硕士生，研究方向为分布式计算和边缘计算等.
李子扬：男，博士，研究员，研究方向为边缘计算、并行计算等.

责任编辑：余 蓉