

星地融合中继网络时延与能耗边缘优化卸载策略

张美楠^{1,2}, 张鸣琪^{1,2}, 丁飞^{1,2}, 庄衡衡^{1,2}, 马海蓉^{1,2}

(1. 南京邮电大学江苏省宽带无线通信和物联网重点实验室, 江苏 南京 210003;

2. 南京邮电大学物联网学院, 江苏 南京 210003)

摘要: 星地融合中继网络 (ISTRN, integrated satellite-terrestrial relay network) 是下一代无线通信系统的必要组成部分, 对加快建成我国空天地一体化网络系统具有重要的现实意义。传统 ISTRN 架构大量的信令需要转发到地面控制中心进行处理, 这增加了网络控制和管理的时间。提出了一种新型云雾计算架构, 在地面接入和中心云之间构建分区域的边缘雾计算层, 提高业务流管理和控制的灵活性。在该云雾网络框架下, 设计了基于 Q -learning 的边缘计算卸载策略, 以时延与能耗作为卸载性能的评价指标。仿真实验结果表明, 与 Min-min 算法和回溯算法相比, 基于 Q -learning 计算卸载算法的时延与能耗性能更优, 且可以在时延与能耗的联合优化之间达成平衡。

关键词: 星地融合中继网络; 多接入边缘计算; 云雾网络; 卸载策略; Q -learning

中图分类号: TP319

文献标志码: A

doi: 10.11959/j.issn.2096-3750.2022.00289

Offloading strategy with edge optimization of time delay and energy consumption in integrated satellite-terrestrial relay network

ZHANG Meinan^{1,2}, ZHANG Mingqi^{1,2}, DING Fei^{1,2}, ZHUANG Hengheng^{1,2}, MA Hairong^{1,2}

1. Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

2. School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Abstract: The integrated satellite-terrestrial relay network (ISTRN) is a necessary part of the next-generation wireless communication system, and has important practical significance for accelerating the construction of my country's air-space-terrestrial integrated network system. In the traditional ISTRN architecture, a large amount of signaling needs to be forwarded to the ground control center for processing, which increases the delay of network control and management. A new cloud fog computing architecture was proposed, which constructs a sub-regional edge fog computing layer between the ground access and the central cloud to improve the flexibility of business flow management and control. Under the cloud network framework, a Q -learning based edge computing offloading strategy was designed, and the offloading performance was evaluated by time delay and energy consumption. Simulation results show that, compared with Min-min algorithm and backtracking algorithm, Q -learning based computational offload algorithm has better performance in terms of time delay and energy consumption, and can achieve a balance between the joint optimization of time delay and energy consumption.

Key words: integrated satellite-terrestrial relay network, multi-access edge computing, cloud fog network, offloading strategy, Q -learning

收稿日期: 2022-03-30; 修回日期: 2022-07-03

通信作者: 丁飞, dingfei@njupt.edu.cn

基金项目: 国家自然科学基金资助项目 (No.61871446, No.61872423); 江苏省重点研发计划 (No.BE2020084-1); 江苏省“六大人才高峰”高层次人才资助项目 (No.DZXX-008); 南京邮电大学科研基金资助项目 (No.NY220028)

Foundation Items: The National Natural Science Foundation of China (No.61871446, No.61872423), The Key Research and Development Program of Jiangsu Province (No.BE2020084-1), The Six Talent Peaks High Level Talent Support Project of Jiangsu Province (No.DZXX-008), The Science Foundation of Nanjing University of Posts and Telecommunications (No.NY220028)

0 引言

我国《“十四五”数字经济发展规划》提出要加快建设高速泛在、天地一体、云网融合、智能敏捷、绿色低碳、安全可控的智能化综合性数字信息基础设施。地面移动网络与卫星通信网络的星地融合网络(STIN, satellite-terrestrial integrated network)是5G和6G网络的重点发展方向之一^[1]。该领域的研究推动了STIN的发展,同时加速驱动6G与广域物联网的创新研究与发展^[2]。3GPP从Release 14(Rel-14)开始研究地面5G与空间星座的通信体制互通的问题^[3],目前,3GPP Rel-18正在推进星地融合增强服务,重点考虑星地双重移动性管理、连续性增强和典型场景服务等需求^[4]。星地融合网络提供增强计算资源、高可靠、低成本、节能的跨域连续通信解决方案,未来将广泛应用于广播与通信、物联网、应急救援等领域^[5]。

由于地面环境感知场景丰富,监测节点异构且部署分散,测量数据种类繁多、信息交换和电信业务多样化^[6],因此在卫星和用户具有双重移动性的场景下,为了更好地满足用户请求和网络资源状态的动态需求,在STIN中保证服务的高连续性和高QoE性能是一个挑战^[7]。为解决上述问题,部署与发展地面中继协助卫星通信的星地融合中继网络(ISTRN, integrated satellite-terrestrial relay network)具有重要的现实意义^[8]。由于ISTRN采用地面中继辅助卫星信号完成数据传输,能够有效提高卫星通信的可靠性和吞吐量,因此发展初期即受到广泛关注^[9-12]。目前,基于DVB-SH标准的ISTRN架构已获得实际应用,文献[13]给出了ISTRN架构的演进方案,包括中继节点设计、服务切换、小区模式生成和资源分配以及改进网络管理和操作等。

在ISTRN中,由于系统异构且分层部署,研究人员将多接入边缘计算(MEC, multi-access edge computing)引入ISTRN系统以增强地面接入与卫星回程业务流的性能,构建具有边缘架构的ISTRN系统并设计业务卸载模型及优化策略^[14]。文献[15]为每个地面站配置一个MEC服务器,设计协同计算卸载模型来实现多任务并行计算,从而优化用户服务质量(QoS, quality of service)。文献[16]提出地面中继和星座分别配置MEC服务器的架构方案,为开展多层边缘计算架构设计和异构边缘计算资源协同调度提供参考;主流卸载框架分为地面用户卸载^[17]、地面站点与卫星之间卸载^[18]以及星间卸载^[19]等。文献[20]为

大规模部署的物联网设备植入了联合任务卸载和资源分配策略,有利于卸载IoRT中计算密集型和时延敏感型任务的高效处理。文献[21]为车联网场景设计了一种计算卸载激励机制,以激励车主允许自身车辆参与到车联网任务的卸载环节,以提升区域资源的利用效率并优化车辆服务质量。

边缘云计算中心架构存在高传输时延的问题,通过雾计算架构可以充分利用边缘云的计算、存储和通信资源优化系统服务时延^[22]。文献[23]提出分布式卫星云雾网络(DSCFN, distributed satellite cloud fog network)架构,由小卫星编队飞行组成卫星雾网络,根据地面站云计算得出的任务划分比例直接进行本地分布式计算,降低业务处理时延。文献[24]研究雾计算网络中边缘计算和卸载问题,提出一种基于联合博弈的解决方案以保证雾节点高效协同工作,从而在短时间内处理大量数据。文献[25]针对云边之间的核心网流量控制问题,提出了关于减少云边通信流量,提高边缘的资源利用率的资源分配和卸载决策算法,从而减少潜在的核心网拥塞。文献[26]针对跨域环境下的多目标资源调度问题,提出了搜索空间“剪枝”的线性加权策略和多目标算法的资源调度方法,可以减少边缘网络的处理时延并且提高网络中设备的能量利用效率,但多目标寻优算法较难获得最优策略。文献[27]提出了一种基于海洋网络连通概率的边缘计算节点选取方法,在近岸场景下使用基于海洋多节点协同卸载遗传算法,在远岸场景下使用基于分组交叉学习粒子群算法,可以促进网络传输时延优化,由于仅考虑了时延优化因素,难以适应节点能量受限场景。在试验环境验证方面,文献[28]搭建了包括1个云数据中心、60个边缘服务器和若干移动终端设备的云雾计算平台,引入深度强化学习实现分区域的卸载处理,可以降低网络服务的时延和能耗等,但该卸载策略依赖于节点和任务之间的关联关系,且计算工作量较大。

上述研究改进了边缘接入与传输过程中任务卸载性能,但大多未能充分考虑能耗因素对系统卸载性能的影响,而这对于ISTRN系统中部署了大量有限能量的接入设备来说是需要重点考虑的。因此,本文将边缘雾计算架构融入ISTRN系统,综合考虑终端、服务器与云雾网络环境,提出一种基于时延和能耗联合优化的任务卸载模型,通过Q-learning算法进行卸载决策优化,确保整体ISTRN系统的服务质量,最后对比不同算法在约束

条件下的卸载性能优劣，找出模型优化设置参数。

1 ISTRN 边缘计算架构

为充分利用星地融合中继网络的计算、存储与通信资源，提出 MEC 赋能的星地融合中继网络，将边缘云靠近用户部署，对汇聚于边缘平台的动态信息按需进行综合处理后，以标准化分级共享的方式支撑不同时延要求下的星地应用需求，达到降低时延且保障用户 QoS 的目标。ISTRN 边缘计算架构如图 1 所示，主要包括地面层、云雾层和卫星层，在地面接入和中心云之间构建分区域的边缘雾计算层，提高业务流管理和控制的灵活性。通过将 MEC 云部署在由多接入终端汇聚的云中心、云雾层的中继服务器以及卫星层的星座，能够保证支持个域网、物联网、互联网、海联网等多场景的泛在接入，并将地面接入数据通过中继站点汇聚，经云雾网络进行分布式计算和卫星雾网络进行协同运作，最终将计算结果直接反馈给目标用户。云雾层以中心云为核心，联合 MEC 赋能的云雾层中继和卫星层星座，形成

多级分布式的云雾计算框架。卫星层由通信、侦查、遥感等多种卫星，或单类型的多颗星座组成，通过协同运作实现星间资源共享利用。

区域云以用户、卫星和环境等实时动态数据为核心，一方面通过卫星地面站实时接收地面层的用户请求，执行业务卸载策略，结合卫星链路状态信息进行数据上传；另一方面实时响应卫星层下发的指令任务。中心云主要负责地面层用户、云雾层中继、卫星层星座的基础数据的集中管理、卫星链路状态监控、星地大数据分析以及卫星层的协作策略管理等。中心云采用雾计算架构，为保障云雾计算网络的性能，通常存在 1 个主雾节点和多个辅助雾节点，而且雾节点的角色可根据设定规则在不同节点之间进行动态调度。

2 系统模型

假设在云雾网络场景中，系统在一次任务卸载过程中有 1 个主雾节点、 N 个终端设备组成集合 $S \in \{S_1, S_2, \dots, S_N\}$ ，以及 M 个辅助雾节点组成雾节点集 $F = \{F_1, F_2, \dots, F_M\}$ ，主雾节点与每个辅助节点之间

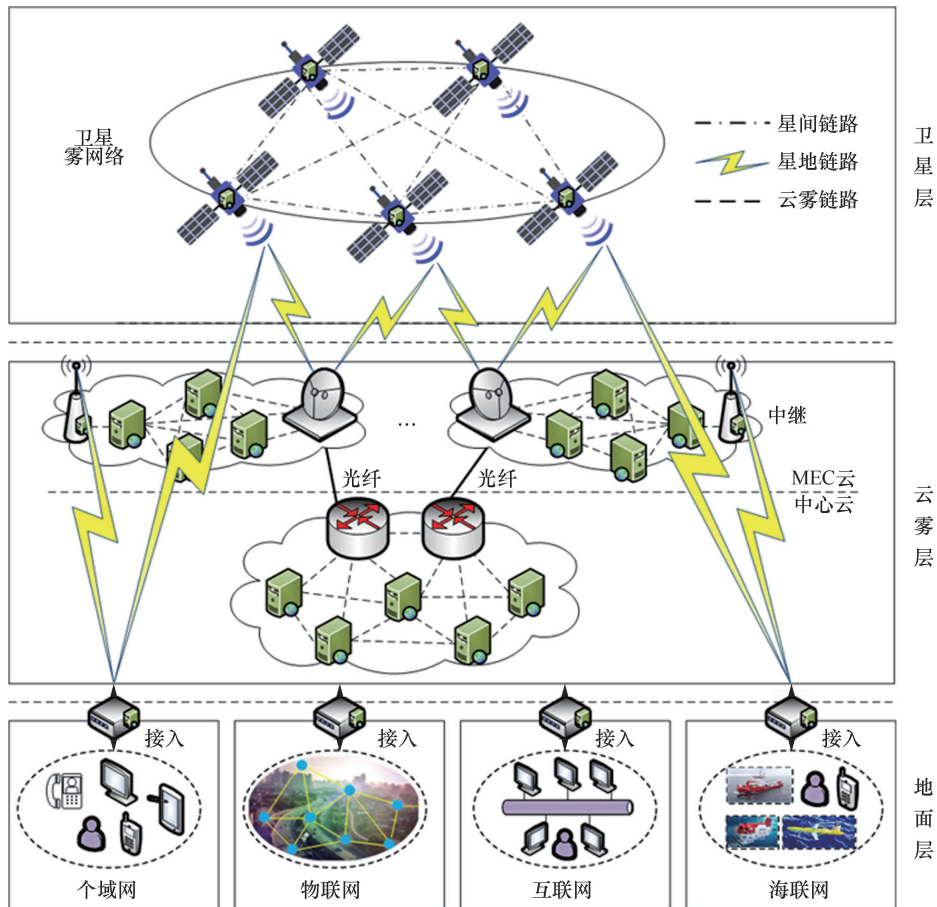


图 1 ISTRN 边缘计算架构

可以实时交互信息。在时隙 t 内, N 个终端设备 $S_i, i \in (1, \dots, N)$ 各自产生的任务 W_i 组成集合 $W = \{W_1, W_2, \dots, W_N\}$, 其随机到达主雾节点且服从泊松分布。任务请求首先发送给雾计算网络中的主雾节点, 主雾节点接收任务请求后依据由主雾节点维护的邻居辅助雾节点列表, 结合卸载算法将任务分发到辅助雾节点进行计算; 最后, 任务在辅助雾节点计算完成后回传给主雾节点, 主雾节点再向终端用户返回计算结果。因此, 在整个任务卸载过程中, 网络的平均服务时延和各个节点的能耗是评价卸载策略的重要性能指标。

2.1 能耗模型

本文研究场景可以描述为: 在时刻 t , 终端用户 S_i 需要解决计算任务集 W , 满足在 M 个辅助雾节点上进行分布式卸载计算。因此, 系统总的能耗包括数据在传输中的通信能耗与雾节点执行计算产生的能耗。

2.1.1 通信能耗

假设系统雾网络环境为无线组网环境, 则通信能耗可以采用一阶无线电模型 (FORM, first order radio model) 表示, 一阶无线电模型由发送能耗、接收能耗和传输能耗组成^[29]。一阶无线电模型参数设置见表 1。

表 1 一阶无线电模型参数设置

参数	描述
e	单位能量消耗常数
μ	发射放大器的单位能量消耗常数
n	传播衰减指数
k	数据量大小/bit
d	数据传输距离

依据 FORM 可以得到: 如果有 k bit 的数据, 从一个雾节点 x 传输到距离为 d 的另一节点 y 的过程中, 产生的通信能耗 $e_{xy}(k, d)$ 可表示为

$$e_{xy}(k, d) = e_t(k, d) + e_r(k) \quad (1)$$

$$e_t(k, d) = e \times k + \mu \times k \times d^n \quad (2)$$

$$e_r(k) = e \times k \quad (3)$$

其中, $e_{xy}(k, d)$ 表示 k bit 的数据从节点 x 传输到距离为 d 的另一节点 y 的通信能耗; $e_t(k, d)$ 表示 k bit 的数据在距离 d 上的发送能耗和传输能耗; $e_r(k)$ 表示 k bit 的数据的接收能耗。

2.1.2 任务计算能耗

根据雾网络系统模型, 假设任务请求集合 $W = \{W_1, W_2, \dots, W_N\}$ 中分别对应 N 个终端各自产生的

计算任务, 每一个终端 S_i 的任务 $W_i, i \in (1, 2, \dots, N)$ 在经过主雾节点分配后, 都有一个对应的辅助雾节点 $F_j, j \in (1, 2, \dots, M)$ 执行其计算任务的需求。因此, 对于单一任务计算所产生的计算能耗 $e(k)$ 可以表示为

$$e(k) = e_j + \rho f_j^2 \tau_{j, W_i} \quad (4)$$

其中, e_j 表示雾节点 F_j 在空闲状态时的能耗; ρ 表示每个雾节点的 CPU 架构常数; f_j 表示雾节点 F_j 的 CPU 频率, CPU 频率越高, 则计算能力越强; ρf_j^2 表示雾节点 CPU 工作一个周期产生的能耗, 该结果越小, 说明该节点性能越好; τ_{j, W_i} 表示雾节点 F_j 完成已分配任务 W_i 所需要的 CPU 周期数。

2.1.3 系统总能耗

结合通信能耗和任务计算能耗计算式, 系统总能耗 E 可表示为

$$E = \sum e_{xy}(k, d) + \sum e(k) \quad (5)$$

其中, $\sum e_{xy}(k, d)$ 表示从节点 x 传输到距离为 d 的节点 y 的过程中, 所有任务的通信总能耗, $\sum e(k)$ 表示所有任务的计算总能耗。

2.2 时延模型

在云雾网络的卸载过程中, 总的服务时延包括数据传输产生的通信时延与任务在雾节点中计算产生的计算时延两部分。

2.2.1 通信时延

假设采用无线信道传输, 且不考虑信道之间的干扰与信道数量的约束。由香农公式可知, 无线传输速率 $C_{x,y}$ 可表示为

$$C_{x,y} = w \log \left(1 + \frac{p_{x,y} H_{x,y}(t)}{N_0} \right) \quad (6)$$

其中, N_0 表示信道噪声功率, w 表示无线信道的带宽, $H_{x,y}(t)$ 表示两节点之间信道的信道增益 (与两节点 x, y 之间的距离和损耗系数有关), N_0 、 w 、 $H_{x,y}(t)$ 均为常数; $p_{x,y}$ 表示节点的发送功率。

由式(6)可知, 任务 W_i 分配到雾计算节点 F_j 产生的通信时延可表示为

$$D_{i,j}^t = \frac{\lambda_i}{C} \quad (7)$$

其中, λ_i 表示完成任务 W_i 的数据量。

2.2.2 计算时延

根据雾节点计算能力^[30], 任务 W_i 分配到节点 F_j 进行计算所需要的计算时间可表示为

$$D_{i,j}^c = \frac{\lambda_i}{f_j} \quad (8)$$

其中, f_j 表示雾节点 F_j 的 CPU 计算频率。

2.2.3 系统总时延

由式(7)和式(8)可知, 完成任务 W_i 系统耗费的总时延 D_j 可表示为

$$D_j = D_{i,j}^t + D_{i,j}^c \quad (9)$$

每个任务在系统中从执行到返回结果所产生的总时延 D 为

$$D = \max \{D_1, D_2, \dots, D_N\} \quad (10)$$

3 基于 Q-learning 的计算卸载算法

考虑到传统 Min-min 算法、回溯算法均为单一约束条件下的优化方法, 无法实现多目标的联合优化。本节引入强化学习求解云雾网络中计算卸载优化策略, 通过对时延与能耗的联合优化分析, 结合对不同业务时延约束的要求, 提升 ISTRN 系统中区域云雾网络计算卸载效率且保障服务质量。

3.1 算法思路与核心要素设计

在 Q-learning 算法中某一时刻 t 下, 首先需要确定的基本要素包括: 环境所处的当前状态 State 记作 s_t 、智能体 Agent (根据不同的环境状态做出的动作, 记作 a_t)、策略 Policy (智能体在当前状态下所做出的一个动作, 记作 $\pi(a_t | s_t)$)、回报 Reward (当状态转移到新的状态时, 环境回馈给智能体的一个奖励, 记作 r_t)、价值函数 Value Function (从当前状态开始, 所有可能的策略及可能转移的状态所获得回报的累计, 记作 $V(s_t)$)、状态价值 State Value (从当前状态执行某个动作后, 获得累计价值的期望, 记作 $Q(s_t, a_t)$)。

Q-learning 计算模型如图 2 所示, 在当前状态为 s_t 时, Agent 在根据现有的经验做出决策动作 a_t 后, 状态会转移到下一个 s_{t+1} , 此时状态发生变化, 且会给智能体一个直接的奖励反馈 r_t 。智能体通过对可选动作的不断尝试、学习、综合评估当前状态到下一时刻的奖励值 r 并累计为价值函数 $V(s_t)$, 从而更新累计价值的期望函数 $Q(s_t, a_t)$, 并使其最大化, 最终找到一个最优的策略。因此, 其适用于求解云雾网络场景下联合优化问题。

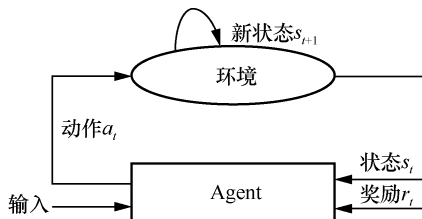


图 2 Q-learning 计算模型

1) 状态空间和动作空间定义

定义: 在雾计算网络系统内, 设主雾节点为 F_0 , M 个辅助雾节点组成的雾节点集合为 $F = \{F_1, F_2, \dots, F_M\}$, 主雾节点可以管理任一辅助雾节点 $F_j, j \in (1, \dots, M)$, 并将 N 个任务组成的集合 $W = \{W_1, W_2, \dots, W_N\}$ 中的任务 $W_i, i \in (1, \dots, N)$ 分配给辅助雾节点执行。

状态 s_t : 在时刻 t 下, 状态空间 S 选取为主雾节点从终端设备获取到的任务集合 W , 每个任务 W_i 都有各自属性。

动作 a_t : 动作空间 A 选取为辅助雾节点集为 F , 主雾节点根据约束条件, 将任务集 W 匹配给辅助雾节点集 F 执行。根据任务执行时间, 获得奖励值。

状态动作价值函数 $Q(s_t, a_t)$ 的转移规则为

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha (r_t(s_t, a_t) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \quad (11)$$

其中, α 是学习效率, 学习效率越大, 收敛越快; γ 为折扣因子, 表示状态回报的影响程度, 折扣因子越大, 则状态回报的影响越大; $\max_{a_{t+1}} Q(s_{t+1}, a_{t+1})$ 表示下一状态 s_{t+1} 中 Q 值最大时的状态动作价值函数值, 并可以得到所采取动作为 a_{t+1} 。

2) 奖励策略

在对某一个策略 $\pi(a_t | s_t)$ 进行评估时, 一般选取立即奖赏价值函数 $R(s_t, a_t)$ 与状态动作价值函数 $Q(s_t, a_t)$ 指标。其中, 立即奖赏价值函数 $R(s_t, a_t)$ 表示在状态 s_t 下, 采取动作 a_t 所取得的立即奖励值 (此处奖励反馈为负反馈, 即时延与能耗越小, 获得的奖励值越大)。相较于传统算法的单一约束条件, Q-learning 算法可以通过对能耗与时延设置一个权值以综合考虑系统性能, 使奖励策略用于描述多目标优化。

通过对环境不断探索, 尝试采取不同的动作并转移到不同的状态, 影响 Q 表不断迭代的过程与奖励值, 最终收敛到累计回报 $V(s_t)$ 的最优解, 表示为

$$V(s_t) = \sum_{i=0}^{\infty} \gamma^i r_{t+i} \quad (12)$$

经过多轮训练之后, 利用已经训练好的 Q 表可以得出最优策略为 π^* , 从而选出在状态 s_t 下, 让价值最大的动作 a_t 。

$$\pi^* = \arg \max V(s_t) \quad (13)$$

算法采用 ϵ -greedy 策略对动作进行选择, 根据

概率值 ε 确定决策过程，当智能体每次做决策时。以 ε 的概率进行探索，即随机选取一个动作；以 $1-\varepsilon$ 的概率进行利用，即选择已有动作中动作价值 Q 最大的动作。在算法的前期，概率 ε 优选较大值，促进 Agent 进行充分探索；随着训练次数的不断增加，概率 ε 不断更新，这样会让 Agent 更多地使用已经学习到的经验进行任务卸载，使得 Q 表收敛速度加快，更快到达收敛效果。

3.2 计算卸载算法

假设当前 Q -learning 智能体所处的状态为 $s_t \in S$ ，其可以选择的动作集合为 A 。 Q -learning 算法在执行某一个动作 $a_t \in A$ 后，将会达到下一个状态 s_{t+1} ，并得到对应的奖励 r_t 。基于 Q -learning 的计算卸载算法见算法 1。episode 代表预先设定的训练周期，利用训练好的 Q 表可以找到最优任务卸载结果，具体实现过程可描述为：

步骤 1 选取初始状态；

步骤 2 根据奖励累积值，选择使 Q 值最大时的动作；

步骤 3 继续下一个状态，回到步骤 1，直到状态空间遍历结束，输出每一步决策下的最终决策策略 π^* ，以此获得主雾节点与多个雾节点之间的卸载策略，主要包括数据流从主雾节点到各个辅助雾节点的数据包的分发路径和对应大小。

在算法 1 中，步骤 1~步骤 2 的复杂度为 $W \times F$ 。步骤 4~步骤 9 的复杂度为 W ，因此步骤 3~步骤 11 的总复杂度为迭代次数的 W 倍。

算法 1 基于 Q -learning 的计算卸载算法

输入：任务集合 $W = \{W_1, W_2, \dots, W_N\}$

雾节点集合 $F = \{F_1, F_2, \dots, F_M, \delta\}$

输出：任务卸载策略 π^*

初始化： $Q(S, A)$

计算单步立即奖励

for W

for F

计算任务 W_i 在节点 F_j 上的处理时延 $d(i, j)$

计算任务处理能耗 $e(i, j)$

计算立即奖励 $r(i, j)$

$$r(i, j) = \frac{1}{\delta d(i, j) + (1 - \delta)e(i, j)}$$

end

end

for episode

```

for  $W$ 
/*确定时刻  $t$  任务的执行状态  $s_t$  和动作  $a_t$ ，以及状态动作价值函数  $Q(s_t, a_t)$  */
依据  $\varepsilon$ -greedy 规则制定策略  $\pi$ ：
生成一个随机数  $x \in (0, 1)$ 
if  $x < \varepsilon$ 
 $a_{t+1} = \text{rand}_j(F)$ 
//随机选取卸载动作
else
 $a_{t+1} = \arg \max Q(s_t, a_t)$ 
//选择可使  $Q(s_t, a_t)$  值最大的动作
计算立即奖励
更新状态动作价值函数  $Q(s_t, a_t)$  的转移规则
end for
end for
输出最优决策  $\pi^*$ 
    
```

4 仿真与结果分析

为评估 ISTRN 系统中采用 Q -learning 算法的计算卸载算法的性能，本文与 Min-min 算法、回溯算法在相同边缘环境条件下的卸载性能进行对比。

4.1 仿真环境与参数设置

仿真实验运行于 Windows 10 操作系统，处理器为 Intel(R) Core(TM) i7-8750H CPU @2.20GHz 2.21 GHz，内存为 8 GB，使用 MATLAB R2019b 软件环境。在仿真过程中，假定 ISTRN 系统中边缘计算场景为 5 km 覆盖范围，生成边缘雾节点网络。边缘雾节点拓扑结构如图 3 所示，其中 1 为主雾节点，其余为辅助雾节点。

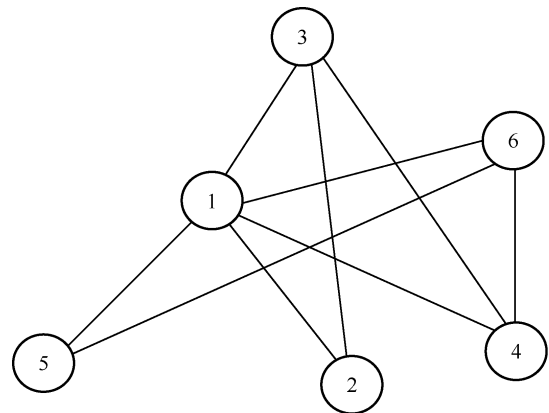


图 3 边缘雾节点拓扑结构

云雾环境与雾节点的仿真参数设置见表 2。

表 2 仿真参数设置

参数	数值
系统带宽	20 MHz
信道噪声功率	-104 dBm
单个任务大小	[10,50] GB
雾节点数	5
雾节点空闲状态能耗	$[1,10] \times 10^{-10}$ J
雾节点 CPU	{100,100,200, 300,150} GHz
CPU 周期数/bit	[500,1 500]周期/bit

4.2 仿真结果分析

在云计算网络环境下，为评估本文提出的基于 *Q-learning* 的计算卸载算法的有效性，将其与 *Min-min* 算法和回溯算法的卸载性能进行对比，通过不同场景下的计算卸载的时延与能耗分析，最后找出时延与能耗联合优化参数。

假配置 5 个辅助雾节点、20 个计算任务的场景，基于 *Q-learning* 的计算卸载训练收敛过程如图 4 所示。可以看出，模型训练迭代到 1 520 次时已进入稳定收敛状态。

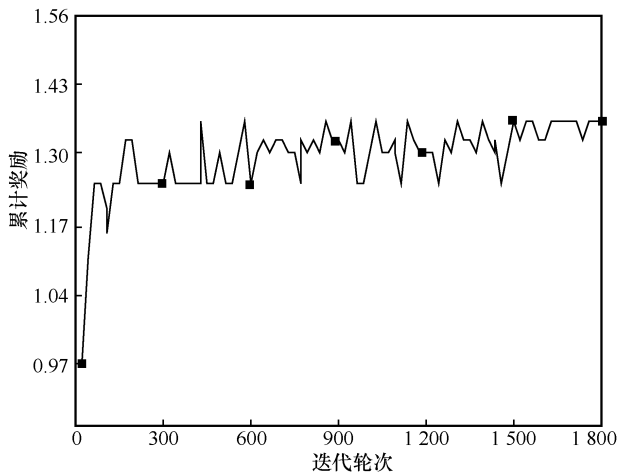


图 4 基于 *Q-learning* 的计算卸载训练过程

4.2.1 系统总时延

不同任务数时的算法运行时间对比如图 5 所示，可以看出，*Min-min* 算法的执行速度最快，变化幅度相对平缓；本文方法在任务数很少时，由于 *Q-learning* 算法的训练周期大，导致运行时间大于其他两种算法，但随着任务数进一步增加，其执行时间明显优于回溯算法；当任务数为 8 个或以内时，*Min-min* 算法和回溯算法的运行时间均小于本文方法；回溯算法在任务数很少时执行速度较快，当任务数超过 9 个后，算法执行时间大幅增加，已经无法满足任务卸载需求；随着任务数继续增加，*Min-min* 算法的运行时间

保持平缓，本文方法的运行时间稍长于 *Min-min* 算法，且保持着较平缓的增长趋势，而回溯算法的算法运行时间随任务数的增加而迅速增加，已无法满足有限资源条件下的优化计算卸载的要求。

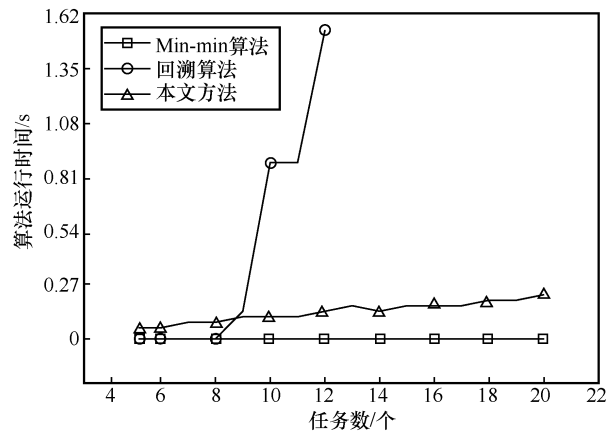


图 5 不同任务数时的算法运行时间对比

不同任务数时的时延性能对比如图 6 表示，可以看出，随着任务数量的不断增加，系统时延均呈上升趋势，其中，*Min-min* 算法的系统时延增长最快，且时延增长趋势存在较大的震荡特性；回溯算法和本文方法在任务数较少时，系统总时延增长趋势基本一致，但当任务数超过 16 个时，回溯算法已无法使系统正常运行，而本文方法的系统时延保持较平稳的增长趋势，即系统时延性能最优。

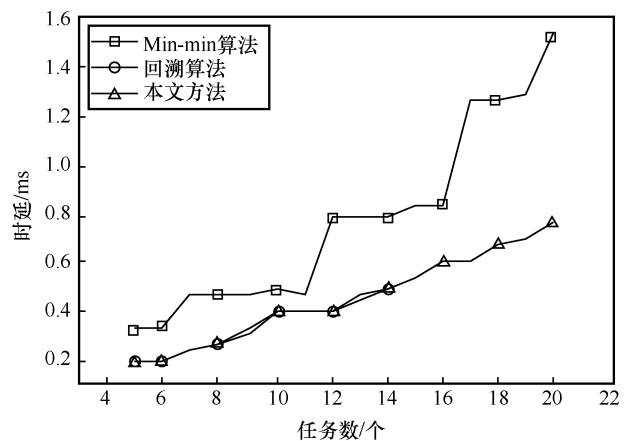


图 6 不同任务数时的时延性能对比

4.2.2 系统总能耗

不同任务数时的系统能耗对比如图 7 所示，可以看出，随着任务数的增加，系统能耗均呈上升趋势。其中，*Min-min* 算法的系统总能耗最大且增长速度最快，且任务数超过 14 个时，系统时延抬升趋势更为显著；回溯算法和本文方法在任务数为 14 个或

以内时, 两者拥有较一致的增长趋势, 但随着任务数的进一步增加, 前者已无法保证系统的正常工作, 后者的系统能耗趋势保持为平缓增长, 即表现出了较好的能耗性能。因此, 相较于传统 Min-min 算法和回溯算法, 本文方法获得的系统能耗性能更优。

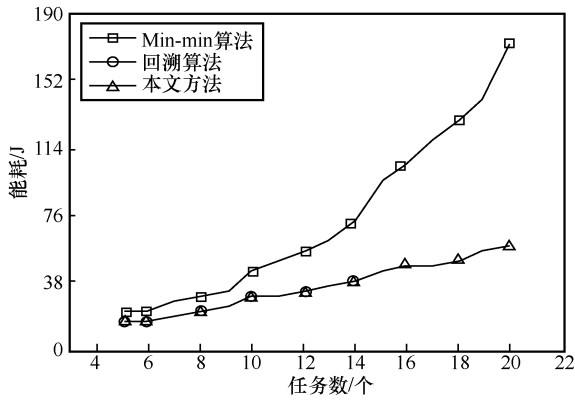


图7 不同任务数时的系统能耗对比

4.2.3 权重因子对系统能耗与时延的影响

考虑5个辅助雾节点和20个计算任务场景, 给时延赋予权重因子 δ , 能耗权重为 $1-\delta$, $\delta \in (0,1)$ 。不同权重因子对时延与能耗的互动影响如图8所示, 实验结果表明, 当 $\delta=1$ 时(即只考虑系统时延), 此时时延最小, 最快完成任务; 当 $\delta=0$ 时(即只考虑系统能耗, 此时能耗最小), 完成任务的能耗最小。当 δ 逐渐增大时, 系统对时延的要求越高, 系统时延逐渐变小, 能耗变高; 当 $\delta < 0.2$ 时, 时延曲线变化幅度较大, 此时提升权重的值, 有利于时延性能的整体提升; 当 $\delta < 0.4$ 时, 能耗曲线变化幅度较小, 能耗增加缓慢; 当 $\delta \geq 0.4$ 时, 能耗曲线幅度增大, 系统能耗较快增加。因此, 在考虑时延与能耗的联合优化时, δ 应选择参数值区间为 $[0.2, 0.4]$, 此时卸载算法的时延与能耗均平缓变化且增长趋势较小, 即系统的整体性能更优。

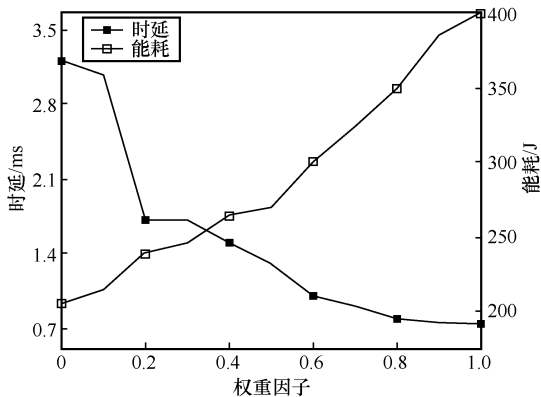


图8 不同权重因子对时延与能耗的互动影响

5 结束语

针对星地融合中继网络带宽资源受限与时延问题, 本文将 MEC 云引入星地融合中继网络架构体系, 提高业务流管理和控制的灵活性。同时, 为云雾组网的 MEC 云设计时延与能耗联合优化的卸载模型, 基于 Q -learning 获得计算任务的最佳卸载策略, 通过时延约束找出联合优化的权重优化参数。仿真结果表明, 所提卸载算法可保证时延且优化能耗。未来可通过对任务优先级进行合理划分, 在满足大规模、不同用户接入服务质量的同时进行卸载决策, 并实现并发异质业务的分布式最优卸载策略。

参考文献:

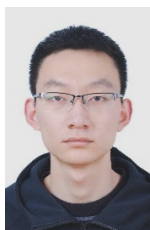
- [1] GUO K F, AN K, ZHOU F, et al. On the secrecy performance of NOMA-based integrated satellite multiple-terrestrial relay networks with hardware impairments[J]. IEEE Transactions on Vehicular Technology, 2021, 70(4): 3661-3676.
- [2] 尤肖虎, 尹浩, 郭贺铨. 6G 与广域物联网[J]. 物联网学报, 2020, 4(1): 3-11.
- [3] YOU X H, YIN H, WU H Q. On 6G and wide-area IoT[J]. Chinese Journal on Internet of Things, 2020, 4(1): 3-11.
- [4] 3GPP. Study on LTE-based V2X services: Release 14 V14.0.0: TR 36.885[S]. 2016-06, [2022-05-10].
- [5] 3GPP. TSG SA priorities[R]. 2022.
- [6] 杨岭才. 关于快速形成我国天地一体通信运营能力的思考[J]. 电信科学, 2022, 38(4): 1-10.
- [7] YANG L C. Thoughts on the rapid formation of China's space-ground integrated communication operation capability[J]. Telecommunications Science, 2022, 38(4): 1-10.
- [8] QIU T, ZHAO Z, ZHANG T, et al. Underwater internet of things in smart ocean: system architecture and open issues[J]. IEEE Transactions on Industrial Informatics, 2020, 16(7): 4297-4307.
- [9] QIAO W X, LU H, LU Y, et al. A dynamic service reconfiguration method for satellite-terrestrial integrated networks[J]. Future Internet, 2021, 13(10): 260.
- [10] AN K, LI Y S, YAN X J, et al. On the performance of cache-enabled hybrid satellite-terrestrial relay networks[J]. IEEE Wireless Communications Letters, 2019, 8(5): 1506-1509.
- [11] EVANS B, WERNER M, LUTZ E, et al. Integration of satellite and terrestrial systems in future multimedia communications[J]. IEEE Wireless Communications, 2005, 12(5): 72-80.
- [12] ZHU X M, JIANG C X, FENGW, et al. Resource allocation in spectrum-sharing cloud based integrated terrestrial-satellite network[C]// Proceedings of 2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC). Piscataway: IEEE Press, 2017: 334-339.
- [13] GUIDOTTI A, VANELLI-CORALLI A, CONTI M, et al. Architectures and key technical challenges for 5G systems incorporating satellites[J]. IEEE Transactions on Vehicular Technology, 2019, 68(3): 2624-2639.
- [14] ZHANG X, JIA M, Gu X, et al. Intelligent spectrum management based on radio map for cloud-based satellite and terrestrial spectrum

- shared networks[EB]. 2021.
- [13] HOSSEINIAN M, CHOI J P, CHANG S H, et al. Review of 5G NTN standards development and technical challenges for satellite integration with the 5G network[J]. IEEE Aerospace and Electronic Systems Magazine, 2021, 36(8): 22-31.
- [14] ALI O, ISHAK M K, BHATTI M K L, et al. A comprehensive review of internet of things: technology stack, middle wares, and fog/edge computing interface[J]. Sensors (Basel, Switzerland), 2022, 22(3): 995.
- [15] ZHANG Z J, ZHANG W Y, TSENG F H. Satellite mobile edge computing: improving QoS of high-speed satellite-terrestrial networks using edge computing techniques[J]. IEEE Network, 2019, 33(1): 70-76.
- [16] XIE R C, TANG Q Q, WANG Q N, et al. Satellite-terrestrial integrated edge computing networks: architecture, challenges, and open issues[J]. IEEE Network, 2020, 34(3): 224-231.
- [17] JI Z, CAO S Z, WU S, et al. Delay-aware satellite-terrestrial backhauling for heterogeneous small cell networks[J]. IEEE Access, 8: 112190-112202.
- [18] DENG R Q, DI B Y, CHEN S Z, et al. Ultra-dense LEO satellite offloading for terrestrial networks: how much to pay the satellite operator? [J]. IEEE Transactions on Wireless Communications, 2020, 19(10): 6240-6254.
- [19] PANG B, GU S S, ZHANG Q Y, et al. CCOS: a coded computation offloading strategy for satellite-terrestrial integrated networks[C]//Proceedings of 2021 International Wireless Communications and Mobile Computing (IWCMC). Piscataway: IEEE Press, 2021: 242-247.
- [20] CHEN T J, LIU J, YE Q, et al. Learning-based computation offloading for IoRT through ka/Q-band satellite-terrestrial integrated networks[EB]. 2021.
- [21] ZHANG X Y, ZHANG H Y, DAI S D, et al. An incentive mechanism for computation offloading in satellite-terrestrial Internet of vehicles[J]. Wireless Communications and Mobile Computing, 2022: 1514437.
- [22] CAO S Z, ZHAO Y, WEI J Y, et al. Space-based cloud-fog computing architecture and its applications[C]//Proceedings of 2019 IEEE World Congress on Services. Piscataway: IEEE Press, 2019: 166-171.
- [23] 任智源, 侯向往, 郭凯, 等. 分布式卫星云雾网络及时延与能耗策略[J]. 浙江大学学报(工学版), 2018, 52(8): 1474-1481.
- REN Z Y, HOU X W, GUO K, et al. Distributed satellite cloud-fog network and strategy of latency and power consumption[J]. Journal of Zhejiang University (Engineering Science), 2018, 52(8): 1474-1481.
- [24] ZHANG G W, SHEN F, LIU Z N, et al. FEMTO: fair and energy-minimized task offloading for fog-enabled IoT networks[J]. IEEE Internet of Things Journal, 2019, 6(3): 4388-4400.
- [25] 李振江, 张幸林. 减少核心网拥塞的边缘计算资源分配和卸载决策[J]. 计算机科学, 2021, 48(3): 281-288.
- LI Z J, ZHANG X L. Resource allocation and offloading decision of edge computing for reducing core network congestion[J]. Computer Science, 2021, 48(3): 281-288.
- [26] LIANG Y, CUI T X, CAO Y, et al. An effective resource scheduling model for edge cloud oriented AIoT[J]. Concurrency and Computation: Practice and Experience, 2022, 34(5): e6720.
- [27] 苏新, 薛溟阳, 周一青, 等. 面向海洋观测监测传感网的计算卸载方法研究[J]. 通信学报, 2021, 42(5): 149-163.
- SU X, XUE H Y, ZHOU Y Q, et al. Research on computing offloading method for oceanview monitoring sensor network[J]. Journal on Communications, 2021, 42(5): 149-163.
- [28] 卢海峰, 顾春华, 罗飞, 等. 基于深度强化学习的移动边缘计算任务卸载研究[J]. 计算机研究与发展, 2020, 57(7): 1539-1554.
- LU H F, GU C H, LUO F, et al. Research on task offloading based on deep reinforcement learning in mobile edge computing[J]. Journal of Computer Research and Development, 2020, 57(7): 1539-1554.
- [29] HEINZELMAN W R, CHANDRAKASAN A, BALAKRISHNAN H. Energy-efficient communication protocol for wireless micro sensor networks[C]//Proceedings of the 33rd Annual Hawaii International Conference on System Sciences. Piscataway: IEEE Press, 2000:10pp.vol.2.
- [30] LYU X C, TIAN H, SENGUL C, et al. Multiuser joint task offloading and resource optimization in proximate clouds[J]. IEEE Transactions on Vehicular Technology, 2017, 66(4): 3435-3447.

[作者简介]



张美楠 (1998-), 女, 南京邮电大学硕士生, 主要研究方向为智慧物联网、星地融合网络。



张鸣琪 (1998-), 男, 南京邮电大学物联网学院在读, 主要研究方向为边缘智能与协同计算技术。



丁飞 (1981-), 男, 博士, 南京邮电大学物联网学院和物联网研究院副教授, 主要研究方向为 5G 网络、星地融合网络、边缘智能与协同计算技术。



庄衡衡 (1998-), 男, 南京邮电大学硕士生, 主要研究方向为物联网与信息系统、边缘智能与协同计算技术。



马海蓉 (1998-), 女, 南京邮电大学硕士生, 主要研究方向为人工智能、星地融合网络。