



空天地协同赋能移动边缘计算: 基于 MASAC 的 URLLC 感知能耗优化

秦鹏*, 李京涵, 伏阳

华北电力大学电气与工程学院, 北京 102206

* 通信作者. E-mail: qinpeng@ncepu.edu.cn

收稿日期: 2025-08-26; 修回日期: 2025-10-23; 接受日期: 2025-11-25; 网络出版日期: 2026-03-16

国家自然科学基金 (批准号: 62201212, 62271201) 和河北省自然科学基金 (批准号: F2025502015) 资助项目

摘要 针对新型电力系统多元主体灵活高效随机接入与服务质量保障需求, 亟需解决现有地面网络通信覆盖不全、任务实时响应弱、设备能量受限等难题. 本文提出了由低轨卫星、无人机和地面用户设备构成的空天地协同赋能移动边缘计算系统模型, 通过研究用户任务切分、卸载策略、功率控制、无人机轨迹规划和边缘计算资源分配的联合优化问题, 以实现终端长期能耗最小化. 然而, 该问题中短期优化决策与长期超高可靠低延迟通信 (ultra reliable and low latency communication, URLLC) 约束高度耦合, 无法直接求解. 为此, 采用李雅普诺夫 (Lyapunov) 优化将原问题转化为一系列可逐时隙求解的子问题. 设计基于多智能体软 Actor-Critic (multi-agent soft actor-critic, MASAC) 的 URLLC 感知能耗优化算法, 并提出了改进的重参数化和激活函数调整策略, 实现对问题的高效求解. 仿真结果表明, 所提方法在降低系统能耗和保障 URLLC 约束方面优于其他基线方法.

关键词 空天地移动边缘计算, 轨迹优化, 资源分配, 超可靠低延迟通信, 多智能体软演员 – 评论家

1 引言

随着信息通信技术的飞速发展和物联网终端的爆炸式增长, 计算密集型及延迟敏感型应用的涌现对无线网络构成严峻挑战^[1]. 以电力垂直行业为例, 大量电力终端部署在远海、戈壁等基础设施薄弱的偏远地区. 这些终端采集的新型电力系统运行数据需要实时传输至远端控制中心, 为风电、光伏等可再生能源的高效消纳提供决策支持. 然而, 现有电力专网和 5G 公网在偏远地区的覆盖能力有限, 且服务质量易受极端天气、地质灾害等突发事件影响, 难以提供稳定可靠的数据传输通道^[2]. 同时, 受限于偏远地区的地理环境, 电力终端普遍采用电池供电且难以频繁更换, 持续的数据传输和计算任务将显著加快终端的能量消耗^[3]. 因此, 亟需构建全域覆盖的新型电力通信网络, 满足多元主体灵活高效的数据传输需求, 并保障海量终端在低功耗条件下的长期稳定运行.

卫星通信系统凭借其广域覆盖、全天候运行等独特优势, 为缺乏地面通信网络覆盖的电力业务场景提供了重要补充. 目前, 北斗卫星已成功支撑偏远地区电力杆塔巡检的短报文传输^[4]; 在郑州“7·20”特大暴雨、超强

引用格式: 秦鹏, 李京涵, 伏阳. 空天地协同赋能移动边缘计算: 基于 MASAC 的 URLLC 感知能耗优化. 中国科学: 信息科学, 2026, 56: 1008–1026, doi: 10.1360/SSI-2025-0368

Qin P, Li J H, Fu Y. Space-air-ground integrated network-empowered mobile edge computing: MASAC-based URLLC-aware energy optimization. Sci Sin Inform, 2026, 56: 1008–1026, doi: 10.1360/SSI-2025-0368

台风“摩羯”等极端场景中,卫星为电力应急抢修指挥提供了关键通信保障^[5].此外,以无人机为代表的空基平台,具有链路视距、部署灵活、成本可控等突出优势,能够在无线网络中发挥覆盖补盲、热点增强的关键作用^[6].由此可见,集成卫星、无人机和地面基础设施构成的空天地协同网络,将全面赋能新型电力系统的信息化能力升级^[7].

以广域同步测量、新能源场站快速功率控制为代表的电力业务均要求毫秒级的时延和99.999%以上的传输可靠性,使得新型电力系统对超可靠和低延迟通信(*ultra reliable and low latency communication, URLLC*)的需求日益迫切^[8].在传统的云计算架构下,终端设备采集的传感数据上传至云端处理将引发较高的端到端时延,无法满足实时电力业务需求.为此,通过在空天地网络节点(即边缘侧)部署计算资源,在靠近数据源处对卸载的任务进行快速处理,是解决该问题的有效技术途径^[9].近年来,随着航天电子技术的飞速发展,星上任务计算处理能力得到了显著增强^[10].我国于2025年5月发射的“三体计算星座”已经搭载星载智算系统,建成后总算力可达1000 POPS,能够为新型电力系统的广域异构数据提供可靠的在轨处理能力.将低轨(*low-earth orbit, LEO*)卫星视为天基边缘节点,利用其低时延特性将部分任务转移至卫星执行也富有前景.因此,如何通过合理的卸载决策、无人机轨迹规划和系统资源配置来充分调用端边算力以确保 URLLC 业务的及时高效处理,是空天地协同赋能移动边缘计算(*mobile edge computing, MEC*)系统亟需研究的问题.

目前已有一些工作研究了空天地协同边缘计算网络的资源分配问题.文献[11]研究了一种新型电力系统星地融合通信架构,采用信息熵理论度量电力数据的紧迫性,并按需激活卫星及地面链路进行传输.文献[12]考虑通过无人机和 LEO 卫星辅助网络偏远地区的电力物联网数据采集,设计了基于聚类和 Q 学习的算法对无人机轨迹和计算资源分配进行联合优化,以最小化平均信息年龄.文献[13]针对电力终端能量受限问题,利用无人机搭载信能同传装置为地面终端提供服务,进而提出了多智能体深度确定性策略梯度(*multi-agent deep deterministic policy gradient, MADDPG*)算法进行轨迹和功率优化.文献[14]将计算任务的持续到达和处理过程建模为队列,并描述了一个长期队列稳定性约束下的数据采集速率最大化问题.作者采用李雅普诺夫(*Lyapunov*)优化和启发式算法进行求解,实现无人机边缘计算资源的安全高效配置.文献[15]提出了一种块连续上界最小化算法,通过同时优化无人机和设备的任务卸载决策来解决空天地协同网络的延迟最小化问题.文献[16]提出了一种基于分布式深度强化学习的算法,优化了空天地协同网络的任务缓存和资源分配,减少了网络传输延迟和天基网络上任务卸载的压力.文献[17]研究了后 5G 远程物联网无人机-低轨卫星集成数据采集的资源分配问题,借助逐次凸逼近和块坐标下降技术,降低系统能耗的同时提高数据采集效率.然而,上述工作关注的都是基于时间平均的服务质量性能度量,如平均延迟、平均能耗、平均队列积压等,仅优化这些平均指标可能导致关键时刻的服务性能不足.例如,排队时延在某一时刻显著高于所容忍的阈值,导致某些任务无法及时处理并造成控制指令的延迟下达,这将对 URLLC 业务造成不良甚至危害性的影响^[18].如何设计超越传统平均性能的优化模型,将 URLLC 约束纳入资源分配之中进而实现电力业务长期精准服务保障,目前还鲜有文献对此进行研究.

对此,本文引入边界违反概率、极端事件发生概率和超额值的统计特性来表征 URLLC 约束.在此基础上,提出一种多智能体软 Actor-Critic (*multi-agent soft actor-critic, MASAC*) 算法来动态平衡 URLLC 约束和能量消耗,实现 URLLC 感知.本文的主要贡献如下.

(1) 提出 LEO 卫星和无人机辅助的空天地协同赋能移动边缘计算系统模型,其中地面用户设备产生的计算任务缓存于队列中,可进一步分为在本地处理和卸载至边缘侧两部分.同时,对用户设备任务缓冲队列积压长度的阈值偏差施加概率约束,并考虑超额值的统计特性构建 URLLC 约束.然后,构建 URLLC 约束下的终端长期能耗最小化问题,并联合优化用户任务切分、卸载策略、功率控制、无人机轨迹规划和边缘计算资源分配.特别地,本文所提 URLLC 约束有效保障了瞬时时延的极端值长期处于规定上限之内,高度契合典型电力业务对于可靠性的严苛要求.

(2) 由于上述问题中短期优化决策与长期能耗及 URLLC 约束高度耦合,导致很难对该问题直接求解.因此,本文利用李雅普诺夫优化将其转化为一系列可逐时隙求解的子问题.进而,本文提出一种基于 MASAC 的 URLLC 感知能耗优化算法.在 MASAC 框架优势的基础上,设计了改进的重参数化及激活函数调整技巧,使其能够输出混合动作变量.所提算法通过连续求解子问题能够动态平衡 URLLC 约束与能量消耗,实现 URLLC 感

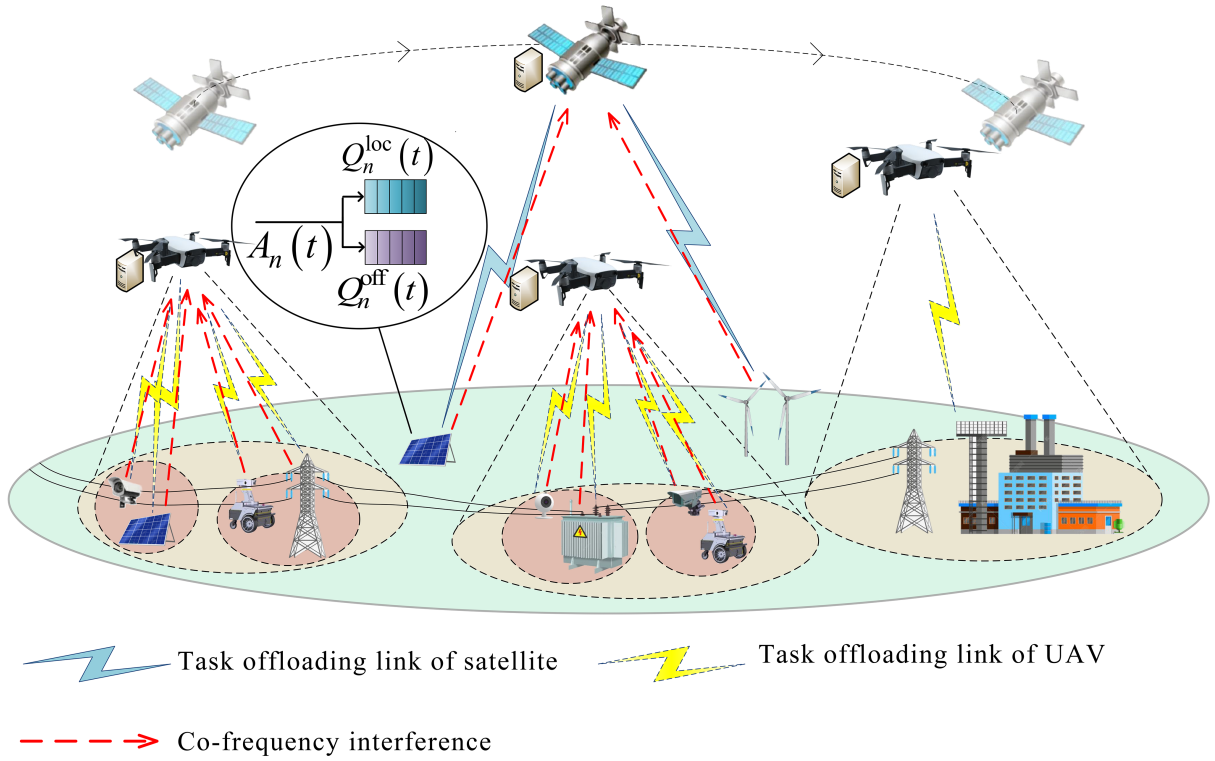


图 1 (网络版彩图) 空天地协同赋能的 MEC 系统模型.

Figure 1 (Color online) Space-air-ground integrated MEC system model.

知能力.

(3) 对所提算法进行了仿真实验与分析. 结果表明, 改进后的 MASAC 算法比基线 DRL 方法 (包括 MADDPG 和 MATD3) 的训练收敛速度更快, 奖励值提升了 9.35%~20.92%. 此外, 与基线 DRL 以及全本地计算、全卸载等方法相比, 本文方法所产生的队列积压降低了 24.9%~31.8%, 系统平均能耗为基准方法的 32.2%~68.9%, 验证了所提方法在平衡系统能耗和任务队列稳定性方面优越的整体性能. 在 URLLC 性能方面, 本文算法以 99.9% 的可靠性保证了任务在限定时间内完成处理. 本研究可为偏远地区低功耗物联终端的高效任务处理提供方案参考.

2 系统模型

如图 1 所示, 本文考虑空天地协同赋能的 MEC 系统, 由一颗 LEO 卫星、多架无人机和大量电力物联网用户设备组成. 空基无人机通过配备边缘服务器为电力用户设备提供计算卸载服务, 按固定轨道移动的天基 LEO 卫星拥有星上处理载荷, 当其经过服务范围时可作为候选的边缘计算节点^[19], 实现基于电力业务的联动服务.

本文充分考虑计算任务的连续到达以及空天地网络节点的动态属性, 研究 T 个时隙的长期系统优化. 时隙集合表示为 $\{1, \dots, t, \dots, T\}$, 单个时隙长度为 τ . 集合 $\mathcal{S} = \{S_0, S_1, \dots, S_k, \dots, S_K\}$ 表示所有空基/天基边缘节点, 其中 S_0 对应卫星 ($k = 0$), $\{S_1, \dots, S_k, \dots, S_K\}$ 对应 K 架无人机. N 个电力用户设备的集合表示为 $\mathcal{U} = \{U_1, \dots, U_n, \dots, U_N\}$. 为充分利用空天地网络资源提升任务处理效率, 允许用户任务切分后在本地处理器和边缘节点并行执行^[20].

鉴于频谱资源的稀缺性, 本文采用 NOMA 技术令多个用户复用相同频段接入边缘节点, 并通过合理的卸载决策与功率控制降低同频干扰. 具体而言, 定义二元卸载指标 $a_n^k(t) \in \{0, 1\}$ 表示第 t 时隙用户 U_n 的卸载决策, 即 $a_n^k(t) = 1$ 表示在第 t 时隙 U_n 选择 S_k 进行任务卸载, 否则 $a_n^k(t) = 0$. 由于每个用户至多选择一架无人机或

卫星进行卸载, 则有约束

$$\sum_{k=0}^K a_n^k(t) \leq 1, \quad \forall n, t. \quad (1)$$

此外, $P_n(t)$ 表示 U_n 的发射功率. 当多个用户选择接入同一边缘节点时, 可通过复用该节点提供的多个正交信道进行任务卸载. 具体而言, 设边缘节点 S_k 提供的信道数为 M^k , 需要分配给当前时隙向其卸载的 N^k 个用户, 用集合 $\{U_1, \dots, U_{N^k}\}$ 表示, 通常 $N^k > M^k$. 因此, S_k 需要将 N^k 个用户分为 M^k 个集群, 进而每个集群采用 NOMA 复用同一正交信道进行任务卸载, 不同集群之间无同频干扰. 本文采用的 NOMA 集群划分方法描述如下, 其核心思想在于最大化同一集群的用户之间信道增益的差距, 从而充分利用 NOMA 的性能增益^[21]. 对于复用边缘节点 S_k 信道 $m \in \{1, \dots, M^k\}$ 的用户, 其信道增益差距计算如下:

$$\Delta g_m^k(t) = \sum_{n=1}^{N_m^k-1} |g_{n+1}^k(t) - g_n^k(t)|, \quad (2)$$

其中, $g_n^k(t)$ 代表 U_n 与 S_k 之间的信道增益, 其具体计算方法将在第 2.2 小节阐述; N_m^k 指复用信道 m 的用户总数. 进而 S_k 所有用户集群的信道增益差距之和表达为

$$\Delta g^k(t) = \sum_{m=1}^{M^k} \Delta g_m^k(t). \quad (3)$$

2.1 无人机运动模型

由于无人机具有部署灵活、强视距概率的特点, 本文同样关注无人机的轨迹优化. 将无人机 S_k 的三维坐标记为 $\delta_k(t) = [x_k(t), y_k(t), z_k(t)]^T$, 其中 $\mathbf{v}_k(t) = [x_k(t), y_k(t)]^T$ 和 $z_k(t)$ 分别表示无人机 S_k 在时隙 t 的水平位置和飞行高度. 假设无人机 S_k 的水平飞行方向角为 $\theta_k(t) \in [0, 2\pi)$, 垂直方向角为 $\phi_k(t) \in [0, \pi]$, 可以得到

$$x_k(t+1) = x_k(t) + l_k(t) \tau \cdot \sin \phi_k(t) \cos \theta_k(t), \quad \forall k, t, \quad (4)$$

$$y_k(t+1) = y_k(t) + l_k(t) \tau \cdot \sin \phi_k(t) \sin \theta_k(t), \quad \forall k, t, \quad (5)$$

$$z_k(t+1) = z_k(t) + l_k(t) \tau \cdot \cos \phi_k(t), \quad \forall k, t, \quad (6)$$

其中, $l_k(t)$ 为无人机的飞行速度. 此外, 无人机运动受如下约束:

$$Z_{\min} \leq z_k(t) \leq Z_{\max}, l_k(t) \leq L^{\max}, \quad \forall k, t, \quad (7)$$

$$0 \leq x_k(t) \leq X_{\max}, 0 \leq y_k(t) \leq Y_{\max}, \quad \forall k, t, \quad (8)$$

其中, $Z_{\min}$ 和 $Z_{\max}$ 分别表示最小高度和最大高度, $L^{\max}$ 为无人机一个时隙内的最大飞行距离, $X_{\max}$ 和 $Y_{\max}$ 为服务区域的边界. 为了避免任意两架无人机之间的碰撞, 无人机间距应不小于最小距离 $D_{\min}$, 即

$$\|\delta_k(t) - \delta_j(t)\| \geq D_{\min}, \quad \forall k, j, k \neq j, \quad \forall t. \quad (9)$$

无人机的推进功率主要受其速度和加速度的影响. 具体来说, 对于速度为 $l_k(t)$ 的旋翼无人机, 其推进功率可以建模为

$$P_k^{\text{fly}}(t) = P_0 \left(1 + \frac{3l_k^2(t)}{U_{\text{tip}}^2} \right) + P_i \left(\sqrt{1 + \frac{l_k^4(t)}{4l_0^4}} - \frac{l_k^2(t)}{2l_0^2} \right)^{\frac{1}{2}} + \frac{1}{2} d_0 \rho s A l_k^3(t), \quad (10)$$

其中, P_0 , P_i , U_{tip} 和 l_0 为与空气动力学相关的常数参数, d_0 和 s 分别表示机体阻力比和转子坚固性, ρ 和 A 分别表示空气密度和转子盘面积.

此外, 在时隙 t , 卫星与用户设备 U_n 之间的距离记为 $d_{n,0}(t)$. 由于 LEO 卫星按固定轨道移动, 因此 $d_{n,0}(t)$ 由卫星的实际位置决定, 不可主动改变.

2.2 通信模型

在无人值守的偏远地区的电力设施所处环境地形复杂, 高程悬殊, 气象多变, 通信信号覆盖不稳定, 因此本文采用概率视距模型建模无人机与电力用户之间的通信链路^[20], 卫星-用户链路则通过自由空间路损模型表征^[22]. 此外, 本文着重考虑了由 NOMA 频谱复用引起的通信干扰.

对于用户-无人机链路, t 时隙从 U_n 到 S_k 的路径损耗表示为

$$L_n^k(t) = 20 \log \left(4\pi f_c \sqrt{z_k^2(t) + r_{n,k}^2(t)/c} \right) + p_{n,k}^{\text{LoS}}(t) \eta_{n,k}^{\text{LoS}} + (1 - p_{n,k}^{\text{LoS}}(t)) \eta_{n,k}^{\text{NLoS}}, \quad (11)$$

其中, $r_{n,k}(t)$ 表示 U_n 和 S_k 之间的水平距离; $\eta_{n,k}^{\text{LoS}}$ 和 $\eta_{n,k}^{\text{NLoS}}$ 分别表示视距 (line-of-sight, LoS) 路径和非视距 (non-LoS, NLoS) 路径的自由空间的加性损耗; f_c 和 c 分别表示载波频率和光速; $p_{n,k}^{\text{LoS}}(t)$ 为空中基站到无人机链路的 LoS 概率, 由下式计算得到:

$$p_{n,k}^{\text{LoS}}(t) = 1/(1 + \mu_1 e^{-\mu_2 [\arctan(z_k(t)/r_{n,k}(t)) - \mu_1]}), \quad (12)$$

这里 $\mu_1, \mu_2, \eta_{n,k}^{\text{LoS}}$ 和 $\eta_{n,k}^{\text{NLoS}}$ 根据环境进行取值. 此后, 定义二者的信道增益为 $g_n^k(t) = 10^{-L_n^k(t)/10}$.

当无人机接收到来自多个用户的信号时, 其采用连续干扰抵消 (successive interference cancellation, SIC) 信号进行解码, 从而降低用户间干扰. 具体而言, 当复用信道 $m \in \{1, \dots, M^k\}$ 的用户 $\{U_1, \dots, U_{N_m^k}\}$ 在时隙 t 向无人机 S_k 卸载时, S_k 对用户的信道增益顺序如下:

$$g_1^k(t) > g_2^k(t) > \dots > g_{N_m^k}^k(t). \quad (13)$$

进而根据该排序进行解码. 因此, 用户 U_n 向 S_k 任务卸载的信干噪比计算为

$$\gamma_n^k(t) = P_n(t) g_n^k / \left(\sum_{i=n+1}^{N_m^k} P_i(t) g_i^k + \rho^2 \right), \quad (14)$$

其中, $\sum_{i=n+1}^{N_m^k} P_i(t) g_i^k$ 表示来自复用 S_k 同一信道 m 的其他用户设备的干扰, ρ^2 表示噪声功率. 定义 B^k 为 S_k 提供的总带宽, 从 U_n 到 S_k 的传输速率可以表示为

$$R_n^k(t) = B^k / M^k \log_2 [1 + \gamma_n^k(t)]. \quad (15)$$

对于用户-卫星链路, 用户 U_n 和卫星之间的信道增益表达为

$$g_n^0(t) = G_0 G_n \varepsilon_0 / d_{n,0}^2(t), \quad (16)$$

其随卫星移动而动态变化, 其中 G_0 和 G_n 分别表示卫星和用户天线增益; ε_0 为参考距离为 1 m 时的信道增益. 同样可采用 SIC 进行解码, 则复用 S_0 的第 m 个信道的用户 U_n 的传输速率为

$$R_n^0(t) = B^0 / M^0 \log_2 \left[1 + P_n(t) g_n^0 / \left(\sum_{i=n+1}^{N_m^0} P_i(t) g_i^0 + \rho^2 \right) \right]. \quad (17)$$

2.3 任务队列模型

本文考虑电力用户生产运营中产生的数据处理任务呈队列形式到达, 并对每个时隙到达的任务进行切分, 部分进行本地计算, 另一部分卸载至边缘节点进行协助计算. 对于用户 U_n , 第 t 个时隙到达的任务量为 $A_n(t)$, 可分为本地任务 $A_n^{\text{loc}}(t)$ 和卸载任务 $A_n^{\text{off}}(t)$, 则有如下约束:

$$A_n(t) = A_n^{\text{loc}}(t) + A_n^{\text{off}}(t), \quad \forall n, t. \quad (18)$$

每个用户有两个缓冲队列, 分别用于存储本地计算任务和卸载任务. 在 t 时隙这两个队列的任务积压用 $Q_n^{\text{loc}}(t)$ 和 $Q_n^{\text{off}}(t)$ 表示, 表达为

$$Q_n^{\text{loc}}(t+1) = \{Q_n^{\text{loc}}(t) - W_n^{\text{loc}}(t)\}^+ + A_n^{\text{loc}}(t), \quad (19)$$

$$Q_n^{\text{off}}(t+1) = \{Q_n^{\text{off}}(t) - W_n^{\text{off}}(t)\}^+ + A_n^{\text{off}}(t), \quad (20)$$

其中, $\{h\}^+$ 是一个指示函数, 即当 $h \geq 0$ 时, $\{h\}^+ = h$, 否则 $\{h\}^+ = 0$; $W_n^{\text{loc}}(t)$ 和 $W_n^{\text{off}}(t)$ 分别为用户本地计算和卸载的任务量, $W_n^{\text{off}}(t)$ 计算为

$$W_n^{\text{off}}(t) = \min \left\{ Q_n^{\text{off}}(t), \tau \sum_{k=0}^K a_n^k(t) R_n^k(t) \right\}. \quad (21)$$

在边缘节点侧, S_k 建立队列 $Q_{n,k}^{\text{edg}}(t)$, 用来存储从 U_n 卸载的任务, 其更新方式为

$$Q_{n,k}^{\text{edg}}(t+1) = \{Q_{n,k}^{\text{edg}}(t) - W_{n,k}^{\text{edg}}(t)\}^+ + A_{n,k}^{\text{edg}}(t), \quad (22)$$

其中, $W_{n,k}^{\text{edg}}(t)$ 为 S_k 边缘计算处理的来自 U_n 的任务量, $A_{n,k}^{\text{edg}}(t) = a_n^k(t) \min \{Q_n^{\text{off}}(t), \tau R_n^k(t)\}$ 为进入队列的任务量.

2.4 计算和能耗模型

本小节分别介绍本地计算和边缘计算模型, 并推导用户能耗. 定义 $f_n(t)$ 为用户设备 U_n 在时隙 t 可用的 CPU 频率, 则 U_n 的本地计算任务量为

$$W_n^{\text{loc}}(t) = \min \{Q_n^{\text{loc}}(t), \tau f_n(t)/\lambda_n\}, \quad (23)$$

其中, λ_n 为所需要计算的密度 (周期/比特). 因此, U_n 在时隙 t 进行本地计算的能耗可计算为

$$E_n^{\text{loc}}(t) = \kappa f_n^3(t) \min \{\lambda_n Q_n^{\text{loc}}(t)/f_n(t), \tau\}, \quad (24)$$

其中, κ 为计算能力参数, 取决于芯片架构^[23].

对于无人机或卫星¹⁾, 其边缘计算的任务量为

$$W_{n,k}^{\text{edg}}(t) = \min \{Q_{n,k}^{\text{edg}}(t), \tau f_{n,k}(t)/\lambda_n\}, \quad (25)$$

其中, $f_{n,k}(t)$ 表示 S_k 分配给来自 U_n 卸载任务的计算资源. 进而, 在时隙 t 用户设备 U_n 将任务卸载至 S_k 的卸载能耗可计算为

$$E_n^{\text{off}}(t) = p_n(t) \sum_{k=0}^K a_n^k(t) \min \{Q_n^{\text{off}}(t)/R_n^k(t), \tau\}. \quad (26)$$

因此, 用户设备在时隙 t 的总能耗表达为²⁾

$$E(t) = \sum_{n=1}^N [E_n^{\text{loc}}(t) + E_n^{\text{off}}(t)]. \quad (27)$$

3 URLLC 约束与问题描述

不同于文献 [24] 中仅考虑任务队列的长期稳定, 本文引入排队时延的 URLLC 约束, 从而确保用户任务的及时处理. 本节首先介绍 URLLC 约束, 并描述能耗优化问题.

1) 针对实际中可能存在的卫星处理时延超出其对终端覆盖时间的情况, 可在卸载决策前基于星历数据估算卫星覆盖时间, 并将即将结束覆盖的卫星排除在候选集外. 同时, 卫星侧可通过星间链路将任务处理结果中继至终端当前可见的卫星, 确保结果有效交付. 由于计算结果的数据量通常远小于原始任务数据 (例如, 在输电线路巡检中, 缺陷定位信息仅为比特量级, 而原始高清影像可达 MB 和 GB 规模), 结果回传的时延通常可忽略不计^[12, 15, 24]. 因此, 本文主要聚焦任务上传与边缘处理阶段的时延及相应的 URLLC 约束. 此外, 得益于低轨卫星较低的轨道高度及其星间激光链路的高传输性能, 通过卫星多跳中继实现结果回传在时延方面具有可行性.

2) 本文考虑在终端侧将任务切分为本地处理和卸载两部分, 后者被传输至 $a_n^k(t) = 1$ 对应的边缘节点 S_k 进行处理. 如果终端盲目选择 $a_n^k(t) = 0$ 以降低传输能耗, 这会导致卸载任务队列 $Q_n^{\text{off}}(t)$ 过度积压, 而不满足后文描述的 URLLC 约束. 因此, 本文通过联合优化任务切分和卸载决策, 旨在保证任务及时处理的同时降低能耗.

3.1 URLLC 约束

与已有研究描述单个任务视角下的边缘计算时延有所不同, 本文考虑用户生成的任务以队列形式持续到达. 因此, 时延约束通过长期 URLLC 约束进行表征. 根据利特尔 (Little) 定律, 任务队列的平均排队时延正比于队列积压与平均数据到达率之比^[23]. 此外, URLLC 约束对排队时延的尾部分布进行限制, 保证可靠性和低延迟. 下面以本地队列 $Q_n^{\text{loc}}(t)$ 为例阐述 URLLC 约束, 其余队列 $Q_n^{\text{off}}(t), Q_{n,k}^{\text{edg}}(t)$ 同理.

首先, 定义 τ_n^{loc} 为 $Q_n^{\text{loc}}(t)$ 的排队时延上限, 则有如下概率约束³⁾:

$$\lim_{T \rightarrow +\infty} \frac{1}{T} \sum_{t=1}^T \Pr \left(\frac{Q_n^{\text{loc}}(t)}{\bar{A}_n^{\text{loc}}(t-1)} > \tau_n^{\text{loc}} \right) \leq \varepsilon_n^{\text{loc}}, \quad (28)$$

其中, $\bar{A}_n^{\text{loc}}(t-1) = \frac{1}{t} \sum_{i=0}^{t-1} A_n^{\text{loc}}(i)$, $\varepsilon_n^{\text{loc}} \ll 1$ 为可容忍的排队时延违反概率.

其次, 需要限制极端事件 $Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}$ 的发生概率. 具体而言, 利用极值理论和 Pickands-Balkema Haan 定理来表征极端事件的统计量和尾部分布^[17]. 令 t 时隙的超额队列积压表示为

$$H_n^{\text{loc}}(t) |_{Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}} = Q_n^{\text{loc}}(t) - \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}} > 0,$$

其分布函数可以用广义帕累托 (Pareto) 分布 $G(h_n^{\text{loc}}; \sigma_n^{\text{loc}}, \xi_n^{\text{loc}})$ 来近似, 其中 σ_n^{loc} 和 ξ_n^{loc} 分别为 G 的尺度参数和形状参数. 因此, $H_n^{\text{loc}}(t)$ 的均值和方差分别表示为

$$\mathbb{E} [H_n^{\text{loc}}(t) | Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}] \approx \sigma_n^{\text{loc}} / (1 - \xi_n^{\text{loc}}), \quad (29)$$

$$\mathbb{D} [H_n^{\text{loc}}(t) | Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}] \approx (\sigma_n^{\text{loc}})^2 / [(1 - \xi_n^{\text{loc}}) (1 - 2\xi_n^{\text{loc}})]. \quad (30)$$

可见随着 σ_n^{loc} 和 ξ_n^{loc} 的增加, $H_n^{\text{loc}}(t)$ 的均值和方差都会增大, 导致更容易出现超额队列积压. 对此, 需要对 σ_n^{loc} 和 ξ_n^{loc} 施加约束, 即 $\sigma_n^{\text{loc}} \leq \sigma_n^{\text{loc,th}}$ 和 $\xi_n^{\text{loc}} \leq \xi_n^{\text{loc,th}}$. 随后, 推导出 $H_n^{\text{loc}}(t)$ 的长期均值和二阶矩的约束如下:

$$\bar{H}_n^{\text{loc}} = \lim_{T \rightarrow +\infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E} [H_n^{\text{loc}}(t) | Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}] \leq \sigma_n^{\text{loc,th}} / (1 - \xi_n^{\text{loc,th}}), \quad (31)$$

$$\begin{aligned} \bar{I}_n^{\text{loc}} &= \lim_{T \rightarrow +\infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E} [I_n^{\text{loc}}(t) | Q_n^{\text{loc}}(t) > \bar{A}_n^{\text{loc}}(t-1) \tau_n^{\text{loc}}] \\ &\leq 2(\sigma_n^{\text{loc,th}})^2 / [(1 - \xi_n^{\text{loc,th}}) (1 - 2\xi_n^{\text{loc,th}})], \end{aligned} \quad (32)$$

其中, $I_n^{\text{loc}}(t) = [H_n^{\text{loc}}(t)]^2$.

最后, 统称 (28), (31) 和 (32) 为本地队列 $Q_n^{\text{loc}}(t)$ 的 URLLC 约束⁴⁾, 统一表达为 $\mathbb{UR} \{Q_n^{\text{loc}}(t)\}$. 进而对于卸载和边缘任务队列同理有 $\mathbb{UR} \{Q_n^{\text{off}}(t)\}$ 和 $\mathbb{UR} \{Q_{n,k}^{\text{edg}}(t)\}$.

3.2 长期能耗最小化问题描述

鉴于地处偏远地区的电力用户设备储能受限, 且不易频繁更换电池, 本文描述一个满足 URLLC 约束的长期用户设备总能耗最小化问题, 旨在联合优化用户任务切分 $\mathbf{A}(t) = \{A_n^{\text{loc}}(t), A_n^{\text{off}}(t) : \forall n\}$ 、卸载决策 $\mathbf{a}(t) = \{a_n^k(t) : \forall n, k\}$ 、功率控制 $\mathbf{P}(t) = \{P_n(t) : \forall n\}$ 、无人机轨迹 $\boldsymbol{\delta}(t) = \{\delta_k(t) : \forall k\}$ 和边缘计算资源分配 $\mathbf{f}(t) =$

3) 该约束要求所有时隙中排队时延超过阈值 τ_n^{loc} 的概率不超过 $\varepsilon_n^{\text{loc}}$, 确保队列积压任务能够在限定时间 τ_n^{loc} 内完成处理, 满足时敏型业务的实时性要求. 高可靠性体现在 $\varepsilon_n^{\text{loc}}$ 通常设定为极小数 (如 10^{-3}), 可实现 URLLC 业务对于“三个九” (99.9%) 高可靠传输的要求, 例如, 电力行业标准 Q/GDW 677 中对分布式电源遥信的可靠性规定^[25].

4) 现有研究中广泛采用的队列稳定性约束 $\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \mathbb{E}[Q(t)] < \infty$ 仅能确保队列积压的长期时间平均有界, 但无法为瞬时排队时延的极端值提供保障^[14, 20, 24]. 相比之下, 本文提出的 URLLC 约束对排队时延的尾部分布进行表征, 并对超额队列积压的高阶统计特性施加明确限制, 从而保证极端事件发生的概率严格低于所容忍的阈值. 与此同时, URLLC 约束引发了更加复杂的长期优化问题描述, 需要在求解时专门考虑.

$\{f_{n,k}(t) : \forall n, k\}$. 该问题写作

$$\begin{aligned}
 \text{P1: } & \min_{\{\mathbf{A}(t), \mathbf{a}(t), \mathbf{P}(t), \boldsymbol{\delta}(t), \mathbf{f}(t) : \forall t\}} \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(t) \\
 \text{s.t. } & C_1: a_n^k(t) \in \{0, 1\}, \forall n, k, t, \\
 & C_2: P_n(t) \leq P_n^{\max}, \forall n, t, \\
 & C_3: P_k^{\text{fly}}(t) \leq P_k^{\text{fly}, \max}, \forall k, t, \\
 & C_4: \sum_{n=1}^N f_{n,k}(t) \leq f_k^{\max}(t), \forall k, t, \\
 & C_5: \mathbb{U}\mathbb{R}\{Q_n^{\text{loc}}(t)\}, \mathbb{U}\mathbb{R}\{Q_n^{\text{off}}(t)\}, \forall n, \\
 & C_6: \mathbb{U}\mathbb{R}\{Q_{n,k}^{\text{edg}}(t)\}, \forall n, k, \\
 & C_7: (1), (4) \sim (9), (18),
 \end{aligned} \tag{33}$$

其中, C_1 代表任务卸载决策为二值变量, C_2 和 C_3 分别表示用户发射功率和无人机推进功率不应超出预算 $P_n^{\max}$ 和 $P_k^{\text{fly}, \max}$, C_4 指出每个边缘节点分配的计算资源不能超出可用资源 $f_k^{\max}(t)$, C_5 和 C_6 代表 URLLC 约束, C_7 表示卸载决策约束 (式 (1))、无人机移动约束 (式 (4)~(9)) 和任务切分约束 (式 (18)).

3.3 基于李雅普诺夫优化的问题转化

在问题 P1 中, 短期的优化决策与长期终端能耗及 URLLC 约束高度耦合并对其产生累积影响, 导致该问题难以直接解决. 为此, 本文采用李雅普诺夫优化^[26] 将问题 P1 转化为一系列可逐时隙求解的子问题. 基于虚拟队列的概念^[27], 将 URLLC 的约束转化为队列稳定性约束. 以 $\mathbb{U}\mathbb{R}\{Q_n^{\text{loc}}(t)\}$ 为例, 可构造如下虚拟队列分别对应 (27), (30) 和 (31):

$$Z_n^{\text{loc},(Q)}(t+1) = \left\{ Z_n^{\text{loc},(Q)}(t) + \mathbb{I}\left\{Q_n^{\text{loc}}(t+1) > \tilde{A}_n^{\text{loc}}(t) \tau_n^{\text{loc}}\right\} - \varepsilon_n^{\text{loc}}, 0 \right\}^+, \tag{34}$$

$$Z_n^{\text{loc},(H)}(t+1) = \left\{ Z_n^{\text{loc},(H)}(t) + \left(H_n^{\text{loc}}(t+1) - \frac{\sigma_n^{\text{loc},\text{th}}}{1 - \xi_n^{\text{loc},\text{th}}} \right) \cdot \mathbb{I}\left\{Q_n^{\text{loc}}(t+1) > \tilde{A}_n^{\text{loc}}(t) \tau_n^{\text{loc}}\right\}, 0 \right\}^+, \tag{35}$$

$$\begin{aligned}
 Z_n^{\text{loc},(I)}(t+1) = & \left\{ Z_n^{\text{loc},(I)}(t) + \left(I_n^{\text{loc}}(t+1) - \frac{2(\sigma_n^{\text{loc},\text{th}})^2}{\left[(1 - \xi_n^{\text{loc},\text{th}}) (1 - 2\xi_n^{\text{loc},\text{th}}) \right]} \right) \right. \\
 & \left. \times \mathbb{I}\left\{Q_n^{\text{loc}}(t+1) > \tilde{A}_n^{\text{loc}}(t) \tau_n^{\text{loc}}\right\}, 0 \right\}^+,
 \end{aligned} \tag{36}$$

其中, h 为真时 $\mathbb{I}\{h\} = 1$, 否则 $\mathbb{I}\{h\} = 0$. 上述三个虚拟队列的意义是每个时隙 t 极端事件的概率、超额队列积压的均值与二阶矩与预定义的阈值之间的偏差, 上标 (Q) , (H) 和 (I) 分别代表与概率、均值和二阶矩对应的标识. 类似地, 可以构造 $\mathbb{U}\mathbb{R}\{Q_n^{\text{off}}(t)\}$ 和 $\mathbb{U}\mathbb{R}\{Q_{n,k}^{\text{edg}}(t)\}$ 的虚拟队列. 通过构建虚拟队列并施加 URLLC 约束, 能够确保排队时延长期稳定在一定阈值范围内, 因而保证了每个时隙到达不同队列的任务都得到及时处理.

基于李雅普诺夫优化, 具有 URLLC 约束的问题 P1 可以转化为寻找总能耗与虚拟队列稳定性的合理折衷. 因此, 将问题转化为在任一时刻 t 最小化如下加权目标函数:

$$\begin{aligned}
 \Theta(t) = & \sum_{n=1}^N \left[\psi_n^{\text{loc}}(t) \cdot \mathbb{I}\left\{Q_n^{\text{loc}}(t+1) > \tilde{A}_n^{\text{loc}}(t) \tau_n^{\text{loc}}\right\} + \psi_n^{\text{off}}(t) \cdot \mathbb{I}\left\{Q_n^{\text{off}}(t+1) > \tilde{A}_n^{\text{off}}(t) \tau_n^{\text{off}}\right\} \right] \\
 & + \sum_{k=0}^K \sum_{n=1}^N \psi_{n,k}^{\text{edg}}(t) \cdot \mathbb{I}\left\{Q_{n,k}^{\text{edg}}(t+1) > \tilde{A}_{n,k}^{\text{edg}}(t) \tau_{n,k}^{\text{edg}}\right\} + VE(t),
 \end{aligned} \tag{37}$$

其中, $\psi_n^{\text{loc}}(t) = Z_n^{\text{loc},(Q)}(t) + Z_n^{\text{loc},(H)}(t) + Z_n^{\text{loc},(I)}(t)$, $\psi_n^{\text{off}}(t) = Z_n^{\text{off},(Q)}(t) + Z_n^{\text{off},(H)}(t) + Z_n^{\text{off},(I)}(t)$, $\psi_{n,k}^{\text{edg}}(t) = Z_{n,k}^{\text{edg},(Q)}(t) + Z_{n,k}^{\text{edg},(H)}(t) + Z_{n,k}^{\text{edg},(I)}(t)$ 代表虚拟队列积压的累加, V 表示控制能耗和虚拟队列稳定的加权因子.

进而, 时隙 t 的短期优化子问题描述为

$$\begin{aligned} \text{P2: } & \min_{\mathbf{A}(t), \mathbf{a}(t), \mathbf{P}(t), \boldsymbol{\delta}(t), \mathbf{f}(t)} \Theta(t) \\ & \text{s.t. } C_1 \sim C_4, C_7. \end{aligned} \quad (38)$$

4 基于 MASAC 的 URLLC 感知能耗优化算法

本节阐述基于 MASAC 的问题 P2 求解算法. SAC 算法已被广泛应用于求解序列决策问题. 与 DQN 和 DDPG 等其他算法相比, SAC 在目标函数中加入了熵正则化项, 提高了探索能力. 然而, SAC 是一种单智能体 DRL 算法, 以集中式的方式输出问题 P2 的高维变量将引发极高的训练复杂度, 同时存在信息集中开销过大的弊端. 为此, 本文采用 SAC 算法的多智能体变体 MASAC 以增强其优化性能^[28]. 下面首先描述马尔可夫 (Markov) 决策过程.

4.1 马尔可夫决策过程描述

在问题 P1 中, 短期的优化决策与长期系统能耗及 URLLC 约束高度耦合并对其产生累积影响, 导致该问题难以直接解决. 我们将 $K+1$ 个边缘节点 (卫星和无人机) 和 N 个用户设备视为 $K+N+1$ 个 MASAC 智能体, 表示为 $\{0, 1, \dots, K, K+1, \dots, K+N\}$, 每个智能体只能根据自己的局部环境信息进行决策. 因此, 马尔可夫决策过程可描述如下.

(1) 观测空间: 边缘节点的观测包含无人机位置、边缘可用计算资源、边缘计算队列及虚拟队列积压, 表达为

$$o_k(t) = \left\{ \boldsymbol{\delta}(t), f_k^{\max}(t), Q_{n,k}^{\text{edg}}(t), Z_{n,k}^{\text{edg}, \Xi}(t) : \Xi \in \{(Q), (H), (I)\}, \forall n \right\}. \quad (39)$$

此外, 用户设备的观测包含无人机的位置、与卫星之间的距离、本地可用计算资源, 以及任务队列和虚拟队列的实时积压, 表达为

$$o_{K+n}(t) = \left\{ \boldsymbol{\delta}(t), d_{n,0}(t), f_n(t), Q_n^{\text{loc}}(t), Q_n^{\text{off}}(t), Z_n^{\text{loc}, \Xi}(t), Z_n^{\text{off}, \Xi}(t) : \Xi \in \{(Q), (H), (I)\} \right\}. \quad (40)$$

因此, 当用户处于多个无人机及卫星覆盖交叉区域时, 其综合考虑可选边缘节点的信道条件和队列积压, 选择最小化能耗同时保证队列稳定的卸载对象.

(2) 动作空间: 在 MASAC 框架下, 每个智能体以分布式的方式输出优化决策, 从而解耦高维动作空间. 具体而言, 边缘节点需要决定其飞行速度、水平方向角和垂直飞行距离, 以及计算资源分配, 其动作表达为

$$\alpha_k(t) = \{l_k(t), \theta_k(t), \phi_k(t), \{f_{n,k}(t) : \forall n\}\}. \quad (41)$$

由于卫星的移动不可自由控制, 可对智能体 $k=0$ 的神经网络输出的飞行相关决策进行裁剪. 用户设备需要选择卸载任务的边缘节点, 并决定发射功率和任务切分, 因此

$$\alpha_{K+n}(t) = \{\{a_n^k(t) : \forall k\}, P_n(t), A_n^{\text{loc}}(t)\}. \quad (42)$$

(3) 奖励函数: 在多智能体协作架构下, 所有智能体获得公共奖励评估联合策略. 奖励设计首先应考虑最小化问题 P2 的目标函数 $\Theta(t)$, 同时应对违反约束的动作进行惩罚. 据此, 公共回报定义为

$$r(t) = -\Theta(t) - \eta_1 \sum_{k=1}^K \zeta_k^b(t) - \eta_2 \sum_{k=1}^K \zeta_k^c(t) - \eta_3 \sum_{k=1}^K \zeta_k^d(t), \quad (43)$$

其中, η_1 , η_2 和 η_3 分别为违反飞行边界、安全距离和推进功率约束的惩罚, $\zeta_k^b(t), \zeta_k^c(t), \zeta_k^d(t) \in \{0, 1\}$ 分别指示无人机 k 是否违反飞行边界、安全距离和推进功率约束.

4.2 MASAC 框架与策略训练

MASAC 算法的目的是最大化长期累积折扣奖励, 同时最大化策略熵, 其学习目标由下式给出:

$$\max_{\pi} \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} [r(t) - \varsigma \log \pi(\boldsymbol{\alpha}(t) | \boldsymbol{o}(t))] \right], \quad (44)$$

其中, $\gamma \in (0, 1)$ 为折扣因子, ς 为温度参数, π 为联合策略, $\boldsymbol{o}(t) = \{o_i(t) : i \in \{0, \dots, K+N\}\}$ 和 $\boldsymbol{\alpha}(t) = \{\alpha_i(t) : i \in \{0, \dots, K+N\}\}$ 分别为所有智能体的观测向量和动作向量, 下文将详细讨论.

在 MASAC 的框架中, 每个智能体都由一个策略网络、两个 Critic 网络及相应的目标网络组成.

(1) 策略网络: 各智能体 i 通过策略网络 π_{φ_i} 根据局部观测 $o_i(t)$ 产生动作 $\alpha_i(t)$, 其中 φ_i 为网络参数. MASAC 框架的策略网络采用随机策略, 即输出动作服从的概率分布, 并通过对概率分布进行采样获得智能体的实际动作, 该方式能极大激励探索. 然而, 采样操作会导致动作不可导, 因此本文采用重参数化技巧来生成动作, 表达为

$$\alpha_i(t) = \alpha_i^{\mu}(t) + \alpha_i^{\sigma}(t) \odot \xi, \quad (45)$$

其中, $\alpha_i^{\mu}(t)$ 和 $\alpha_i^{\sigma}(t)$ 分别为策略网络输出的动作均值和方差, ξ 是标准正态分布随机数, $\odot$ 指哈达玛 (Hadamard) 积.

(2) Critic 网络: Critic 网络用于拟合智能体的软 Q 函数, 即对学习目标 (44) 的估计. 在集中训练过程中, Critic 网络能够获得所有智能体的观测向量 $\boldsymbol{o}(t)$ 和动作向量 $\boldsymbol{\alpha}(t)$ 作为输入, 并估计软 Q 值. 为了减轻对软 Q 值的高估, 任一智能体 $i \in \{0, \dots, K+N\}$ 采用双 Critic 网络 $Q_{\theta_{i,1}}, Q_{\theta_{i,2}}$ 及其对应的目标 Critic 网络 $Q_{\theta'_{i,1}}, Q_{\theta'_{i,2}}$, 网络参数分别为 $\theta_{i,1}, \theta_{i,2}$ 和 $\theta'_{i,1}, \theta'_{i,2}$. 此后, 智能体在每个时隙与环境的交互可生成经验元组 $(\boldsymbol{o}(t), \boldsymbol{\alpha}(t), r(t), \boldsymbol{o}(t+1))$, 并放入重放记忆缓冲区 B 中. 在训练过程中, 神经网络的参数可以根据从重播内存缓冲区 B 中采样小批经验元组 B 进行优化, 其中 $B \in B$, 小批个数为 $|B|$.

根据 MASAC 策略训练, 各智能体 i 的 Critic 网络参数的更新需要最小化如下损失函数:

$$J(\theta_{i,j}) = \frac{1}{|B|} \sum_{b \in B} (y_i(b) - Q_{\theta_{i,j}}(b))^2, \quad j \in \{1, 2\}, \quad (46)$$

其中, $y_i(b)$ 为 Critic 网络的目标值, 可以表示为

$$y_i(b) = r(b) + \gamma (\min_{j=1,2} Q_{\theta'_{i,j}}(b) - \varsigma \log \pi_{\varphi_i}(\alpha_i(b+1) | o_i(b+1))). \quad (47)$$

此外, 温度参数 ς 表征奖励项与熵项的相对重要性, 其自适应调整对平衡奖励最大化和探索至关重要, ς 的更新误差为

$$J(\varsigma) = \frac{1}{|B|} \sum_{b \in B} [-\varsigma \log \pi(\alpha(b) | \boldsymbol{o}(b)) - \varsigma H_0], \quad (48)$$

其中, H_0 为预定义的最小熵. 各智能体 i 的策略网络参数根据下式进行更新:

$$J(\phi_i) = \frac{1}{|B|} \sum_{b \in B} \left[\varsigma \log \pi_{\phi_i}(\alpha_i(b) | o_i(b)) - \min_{j=1,2} Q_{\theta_{i,j}}(b) \right]. \quad (49)$$

目标 Critic 网络由下式进行更新:

$$\theta'_{i,j} \leftarrow \iota \theta_{i,j} + (1 - \iota) \theta'_{i,j}, \quad i = 1, 2, \quad (50)$$

其中, ι 为软更新参数. 特别地, 考虑到问题 P2 存在一系列对优化变量的约束, 因此还需要在 MASAC 框架的基础上设计改进的约束保障技巧. 其一, C_1 限制了卸载决策变量为二值, 但 MASAC 仅适用于输出连续动作. 为此, 本文引入 Gumbel-Softmax 重参数化对各用户智能体 n 的策略网络输出的连续卸载决策变量 $\{\tilde{a}_n^k(t) : \forall k\}$ 执行如下变换:

$$a_n^k(t) = \frac{\exp[(U_n^k + \log \tilde{a}_n^k(t))/s_0]}{\sum_{l=0}^K \exp[(U_n^l + \log \tilde{a}_n^l(t))/s_0]}, \quad \forall k, \quad (51)$$

其中, U_n^k 服从概率密度为 $\exp(-u - \exp(-u))$ 的 Gumbel 分布, s_0 为 Softmax 逼近因子. 其二, 用户发射功率、任务切分、无人机飞行速度和角度变量都存在上限, 可添加激活函数 Sigmoid 将策略网络输出限制在 $[0, 1]$, 进而乘以上限值得到实际决策变量. 其三, 边缘节点智能体 k 的计算资源分配约束 C_3 可通过激活函数 Softmax 来确保, 即

$$f_{n,k}(t) = \frac{\exp[\tilde{f}_{n,k}(t)]}{\sum_{n=1}^N \exp[\tilde{f}_{n,k}(t)]} f_k^{\max}(t), \quad \forall n. \quad (52)$$

通过上述策略对 MASAC 框架进行改进后, 可使其良好适配本文考虑的空天地协同赋能 MEC 环境, 从而高效解决问题 P2.

4.3 基于 MASAC 的 URLLC 感知能耗优化算法

综上所述, 本文提出基于 MASAC 的 URLLC 感知能耗优化算法, 如算法 1 所示. 首先, 所有智能体与环境进行交互收集历史经验, 同时策略网络输出的动作需要通过所提约束保障技巧进行处理. 其次, 通过最小化损失函数对所有智能体的 Critic 网络和策略网络进行训练. 最后, 训练完备的策略网络能够仅基于局部观测执行分布式决策, 用户设备与边缘智能体相互协作来降低能耗并控制与 URLLC 约束相关的虚拟队列积压, 从而实现 URLLC 感知.

下面给出算法 1 收敛性和队列稳定性的理论证明.

Algorithm 1 MASAC-based URLLC-aware energy optimization algorithm.

Input: Space-air-ground integrated network-empowered MEC environment, neural network structure, and hyperparameters of MASAC;

```

1: Initialized weights  $\{\theta_{i,1}, \theta_{i,2}, \theta'_{i,1}, \theta'_{i,2}, \varphi_i : \forall i\}$ , experience replay buffer;
2: while  $ep < ep_{\max}$  do
3:   while  $t \leq T$  do
4:     for Each agent  $i$  do
5:       Obtain observations using (39), (40);
6:       Output actions using policy network  $\pi_{\varphi_i}$ ;
7:       Obtain real actions  $\alpha_i(t)$  using constraint guarantee tricks;
8:     end for
9:     Update task queue and virtual queue backlogs;
10:    Calculate reward using (43) and transit to the next state;
11:    Store  $(\mathbf{o}(t), \boldsymbol{\alpha}(t), r(t), \mathbf{o}(t+1))$  into  $B$ ;
12:    for Each agent  $i$  do
13:      Randomly sample  $B$  experience tuples;
14:      Update critic network weights  $\theta_{i,1}, \theta_{i,2}$  by minimizing (46);
15:      Update policy network weights  $\varphi_i$  by minimizing (49);
16:      Update target network weights  $\theta'_{i,1}, \theta'_{i,2}$  using (50);
17:    end for
18:    Update temperature parameters by minimizing (48);
19:  end while
20: end while
Output: Well-trained policy network  $\{\pi_{\varphi_i} : \forall i\}$ .

```

(1) 收敛性: 算法 1 的收敛性证明分为如下三个步骤. 首先, 计算所有智能体的联合策略 $\pi_{\phi}(\boldsymbol{\alpha} | \mathbf{o}) =$

$\prod_{i=0}^{K+N} \pi_{\phi_i}(\alpha_i | o_i)$ 的梯度如下:

$$\begin{aligned} g_{\pi} &= \mathbb{E} \left\{ \sum_{i=0}^{K+N} \nabla_{\phi} \left[\varsigma \log \pi_{\phi_i}(\alpha_i | o_i) - \min_{j=1,2} Q_{\theta_{i,j}}(\mathbf{o}, \boldsymbol{\alpha}) \right] \right\} \\ &= \mathbb{E} [\nabla_{\phi} (\varsigma \log \pi(\boldsymbol{\alpha} | \mathbf{o}) - Q(\mathbf{o}, \boldsymbol{\alpha}))], \end{aligned} \quad (53)$$

其中, $\phi = \{\phi_i : \forall i\}$, $Q(\mathbf{o}, \boldsymbol{\alpha})$ 代表全局软 Q 值. 由于所有智能体相互协作以获取最大全局回报, 式 (53) 与单智能体 SAC 的标准策略梯度具有相同形式. 其次, 定义 π^{old} 表示原始策略, π^{new} 为根据式 (53) 进行梯度下降更新后的策略, 则有

$$\mathbb{E}_{\boldsymbol{\alpha} \sim \pi^{\text{new}}} [\varsigma \log \pi^{\text{new}}(\boldsymbol{\alpha} | \mathbf{o}) - Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha})] \leq \mathbb{E}_{\boldsymbol{\alpha} \sim \pi^{\text{old}}} [\varsigma \log \pi^{\text{old}}(\boldsymbol{\alpha} | \mathbf{o}) - Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha})]. \quad (54)$$

列写关于 $Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha})$ 的软贝尔曼 (Berman) 方程如下:

$$\begin{aligned} Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) &= r(\mathbf{o}, \boldsymbol{\alpha}) + \gamma \mathbb{E}_{\boldsymbol{\alpha}'} [\mathbb{E}_{\boldsymbol{\alpha}' \sim \pi^{\text{old}}} [Q^{\pi^{\text{old}}}(\mathbf{o}', \boldsymbol{\alpha}') - \varsigma \log \pi^{\text{old}}(\boldsymbol{\alpha}' | \mathbf{o}')]] \\ &\leq r(\mathbf{o}, \boldsymbol{\alpha}) + \gamma \mathbb{E}_{\boldsymbol{\alpha}'} [\mathbb{E}_{\boldsymbol{\alpha}' \sim \pi^{\text{new}}} [Q^{\pi^{\text{old}}}(\mathbf{o}', \boldsymbol{\alpha}') - \varsigma \log \pi^{\text{new}}(\boldsymbol{\alpha}' | \mathbf{o}')]] \\ &\triangleq \mathcal{T}^{\pi^{\text{new}}} Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}), \end{aligned} \quad (55)$$

其中 $\mathcal{T}^{\pi^{\text{new}}}$ 表示基于新策略 π^{new} 对软 Q 值进行更新的算子. 进而, 根据 (55) 对 $Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha})$ 重复作用该算子可得 $Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \leq \mathcal{T}^{\pi^{\text{new}}} Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \leq \mathcal{T}^{\pi^{\text{new}}} (\mathcal{T}^{\pi^{\text{new}}} Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha})) = (\mathcal{T}^{\pi^{\text{new}}})^2 Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \leq \dots \leq (\mathcal{T}^{\pi^{\text{new}}})^m Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \leq \dots$. 当 $m \rightarrow \infty$ 时, $(\mathcal{T}^{\pi^{\text{new}}})^m Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \rightarrow Q^{\pi^{\text{new}}}$, 因此有 $Q^{\pi^{\text{old}}}(\mathbf{o}, \boldsymbol{\alpha}) \leq Q^{\pi^{\text{new}}}(\mathbf{o}, \boldsymbol{\alpha})$, 这表示根据 (53) 更新的策略将提升全局软 Q 值. 最后, 证明迭代收敛的策略 (参数用表示 ϕ^*) 具有最小的误差函数, 即 $J(\phi^*) < J(\phi)$, $\forall \phi \neq \phi^*$, 这是因为收敛的策略具有最大的软 Q 值. 综上可得, 算法 1 必然收敛至局部最优策略, 该策略满足

$$\liminf_{t \rightarrow \infty} \|\nabla_{\phi} J(\phi)\| = 0. \quad (56)$$

此外, 算法 1 以分布式的方式求解问题 P2, 其通过智能体与环境的交互学习最大化累计回报的策略, 允许智能体之间共享观测与动作来解耦动作空间, 同时通过本文设计约束保障技巧来增强 MASAC 的收敛性.

(2) 队列稳定性: 算法 1 能够有效保障虚拟队列的稳定性, 并且能耗最小化的目标与队列稳定性之间存在折衷关系. 具体而言, 本文基于李雅普诺夫优化理论来处理 URLLC 约束, 从而将问题 P1 转化为问题 P2. 以本地计算的虚拟队列 $\mathbf{Z}(t) = \{Z_n^{\text{loc}, \Xi}(t) : \Xi \in \{(Q), (H), (I)\}, \forall n\}$ 为例构造

$$L(\mathbf{Z}(t)) = \frac{1}{2} \sum_{n=1}^N \sum_{\Xi \in \{(Q), (H), (I)\}} (Z_n^{\text{loc}, \Xi}(t))^2.$$

则漂移函数表示为

$$\Delta(t) = \mathbb{E}[L(\mathbf{Z}(t+1)) - L(\mathbf{Z}(t)) | \mathbf{Z}(t)] \leq C + \sum_{n=1}^N \sum_{\Xi \in \{(Q), (H), (I)\}} Z_n^{\text{loc}, \Xi}(t) \mathbb{E}[A_n^{\Xi}(t) - B_n^{\Xi}(t) | \mathbf{Z}(t)],$$

其中 $A_n^{\Xi}(t)$ 和 $B_n^{\Xi}(t)$ 分别为 t 时隙队列积压的增加量和减少量, 可通过式 (34)~(36) 获得, C 为常数. 假设问题 P1 存在一个可行的稳定解使得 $\mathbb{E}[A_n^{\Xi} - B_n^{\Xi}] \leq -\epsilon, \epsilon > 0$, 对应的最优能耗为 E^* , 而本文通过求解问题 P2 来最小化每个时隙的加权函数 $\Theta(t)$, 因此有

$$\Delta(t) + V \mathbb{E}[E(t) | \mathbf{Z}(t)] \leq C - \epsilon \sum_{n=1}^N \sum_{\Xi \in \{(Q), (H), (I)\}} Z_n^{\text{loc}, \Xi}(t) + V E^*. \quad (57)$$

对上式取期望并对 $t = 1, \dots, T$ 求和, 可得

$$\mathbb{E}[L(\mathbf{Z}(T+1)) - L(\mathbf{Z}(1))] + V \sum_{t=1}^T \mathbb{E}[E(t)] \leq CT - \epsilon \sum_{t=1}^T \sum_{n=1}^N \sum_{\Xi \in \{(Q), (H), (I)\}} Z_n^{\text{loc}, \Xi}(t) + V T E^*. \quad (58)$$

表 1 场景参数表.

Table 1 Scenario parameter table.

Parameter	Value	Parameter	Value
Number of time slots T	200	Slot duration τ	0.1 s ^[20]
Number of UAVs K	3	Number of devices N	15
UAV altitude range $Z_{\min}, Z_{\max}$	50, 100 m ^[21]	Satellite altitude z_0	550 km ^[12]
Numer of sub-channels M^k	1	$f_c, \eta_{n,k}^{\text{LoS}}, \eta_{n,k}^{\text{NLoS}},$	0.1 GHz, 0.1, 2.1 ^[20]
Bandwidth of edge nodes B_k	1 MHz	μ_1, μ_2, G_0, z_0	4.88, 0.43, -30 dB, 1000 km ^[12]
Maximum device power $P_n^{\max}$	23 dBm ^[14]	Noise power ρ^2	-144 dBm ^[20]
Edge capability $f_k^{\max}(t)$	[79.2, 118.8] GHz ^[20]	Flight energy $E_k^{\text{fly}, \max}$	2 kJ ^[21]
Local capability $f_n(t)$	[1.6, 2.4] GHz ^[20]	Computing density λ_n	1000 CPU cycles/bit ^[20]
$\tau_n^{\text{loc}}, \varepsilon_n^{\text{loc}}, \sigma_n^{\text{loc}, \text{th}}, \xi_n^{\text{loc}, \text{th}}$	0.3, 10^{-3} , 16×10^3 , 0.27 ^[18]	Task data size $A_n(t)$	[24, 36] kbits ^[13, 20, 22]
		Computing parameter κ	5×10^{-26} J/(Hz ³ · s) ^[20]

表 2 算法参数表.

Table 2 Algorithm parameter table.

Parameter	Value	Parameter	Value	Parameter	Value	Parameter	Value
Weight V	1×10^8	Discount factor γ	0.95	Initial temperature ς	0.2	Episode number	500
Learning rate	0.0025	Soft update rate ι	0.005	Batch size $ B $	256	Hidden neuron number	256

整理不等式并令 $T \rightarrow \infty$, 同时考虑到 $L(\mathbf{Z}(1))$ 是由初始状态决定的有限值, 可以得到

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \sum_{n=1}^N \sum_{\Xi \in \{(Q), (H), (I)\}} Z_n^{\text{loc}, \Xi}(t) \leq \frac{C + V(E^* - \frac{1}{T} \sum_{t=1}^T \mathbb{E}[E(t)])}{\epsilon} < \infty. \quad (59)$$

这表明虚拟队列是处于长期的稳定状态. 上式还指出问题 P2 的最优解 $\frac{1}{T} \sum_{t=1}^T \mathbb{E}[E(t)]$ 以 $o(1/V)$ 渐近问题 P1 的最优解 E^* , 因此提升 V 能够降低系统能耗以缩小最优间隙, 但会增加违反 URLLC 约束的概率.

5 仿真结果与分析

本节通过仿真实验来评估我们的解决方案. 选择一个 $1000 \text{ m} \times 1000 \text{ m}$ 的正方形区域, 其中用户呈随机分布. 仿真用到的场景参数和算法参数分别见表 1 和 2, 参数设置基于文献 [12~14, 20~22]. 为了评估所提算法的性能, 本文与如下 4 种基准方法进行比较.

- (1) MATD3^[29]: TD3 算法的多智能体变体, 采用确定性策略梯度, 依赖添加人工噪声进行探索.
- (2) MADDPG^[30]: DDPG 算法的多智能体变体算法, 相比 MATD3 缺少双 Q 网络等框架改进.
- (3) 全卸载方法: 用户设备的任务全部卸载至边缘服务器计算, 其他变量按所提算法进行优化.
- (4) 全本地计算方法: 用户设备产生的任务全部进行本地计算, 其他变量按所提算法进行优化.

图 2 显示了所有智能体相对于整个训练过程的累计奖励. 可以观察到, 在训练阶段开始时, 曲线的波动率增加. 这是因为每个终端智能体可以根据不同的观察结果 (包括任务到达和无人机移动) 选择更好的任务卸载候选者. 当所有智能体都找到合适的策略时, 训练曲线收敛, 但由于任务到达和计算资源的动态变化, 训练曲线会出现波动. 与 MATD3 和 MADDPG 方法相比, 本文 MASAC 方法具有更高的收敛速度 (约提前了 40 个回合达到收敛), 奖励值提升了 9.35%~20.92%.

图 3 和 4 为无人机轨迹图, 其中有 3 架无人机和 12 个用户. 从图 3 可以看到, 每架无人机能够从其初始位置依次飞过其关联的用户终端. 这是因为通过 MASAC, 无人机和用户终端分别学到了合适的飞行轨迹和任务卸载策略, 用户终端将任务卸载至合适的无人机进行辅助计算. 此外, 从图 4 可以看到, UAV 在从初始位置出发

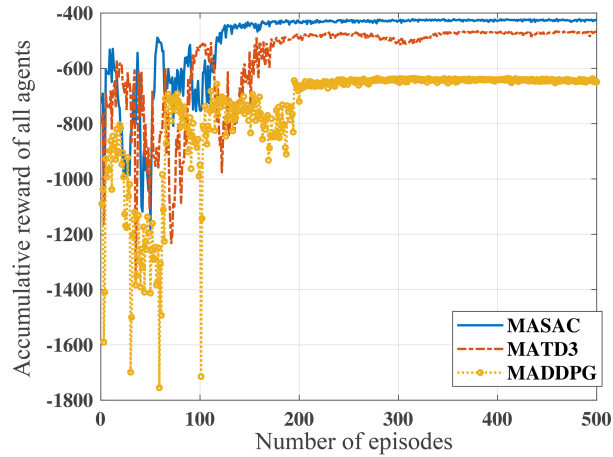


图2 (网络版彩图) 不同算法奖励总和随回合数变化.

Figure 2 (Color online) Accumulative rewards for different algorithms versus episode number.

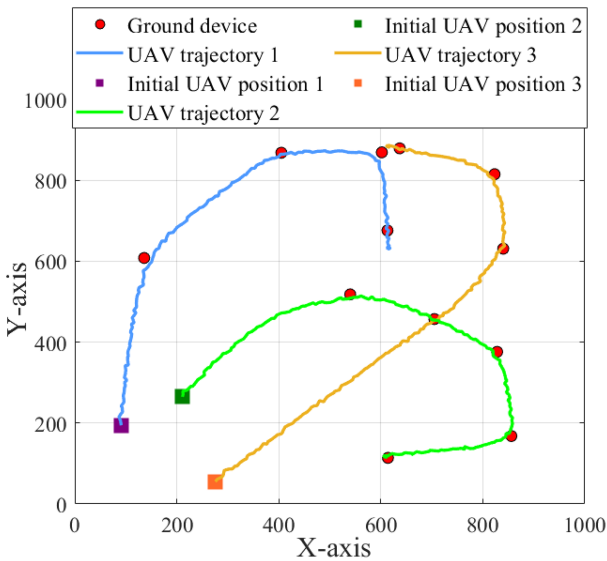


图3 (网络版彩图) 2D UAV 轨迹图.

Figure 3 (Color online) 2D UAV trajectory map.

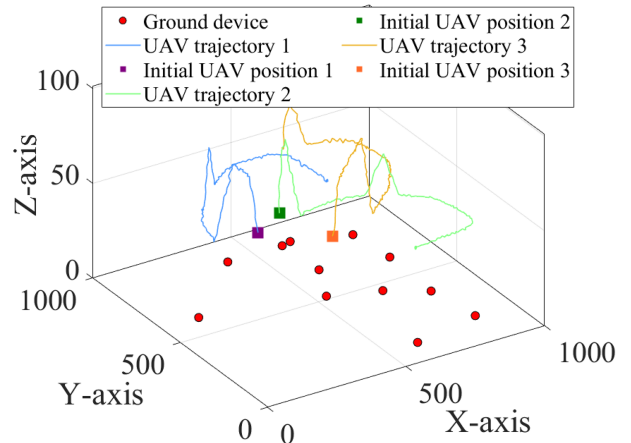


图4 (网络版彩图) 3D UAV 轨迹图.

Figure 4 (Color online) 3D UAV trajectory map.

时会逐渐升高, 而接近服务的用户终端设备时, 其高度会降低. 这是为了通过适当控制高度和仰角来保持较大的视距概率, 从而提高信道功率增益.

图5~7展示了不同算法下各类队列积压随时隙变化的情况. 在图5中, 全本地计算方法产生了最大的本地计算平均队列积压, 这是因为所有任务都由用户终端进行本地计算, 而用户终端计算资源有限, 因此无法及时处理本地计算队列中的任务. 在图6和7中, 全卸载方法产生了最大的卸载队列积压和无人机队列积压, 这是因为用户终端将所有产生的任务全部卸载至无人机, 而传输功率和带宽资源有限, 因此任务在卸载队列和无人机队列中无法得到及时处理. 可以看到, 所有算法中, 随着任务的持续到达和处理, 本文方法对应的本地计算、任务卸载和边缘计算队列积压都达到了一个较低的稳定值, 比其他基准方案降低了24.9%~31.8%, 展现出了良好的队列稳定性能.

图8为系统平均能耗随时隙变化图, 其中全本地计算方法产生了最大的系统能耗. 这是因为将所有任务全部由用户设备处理将会在计算过程中产生大量能耗. 而全卸载方法则产生了最低的能耗, 但如前所述, 该方法会

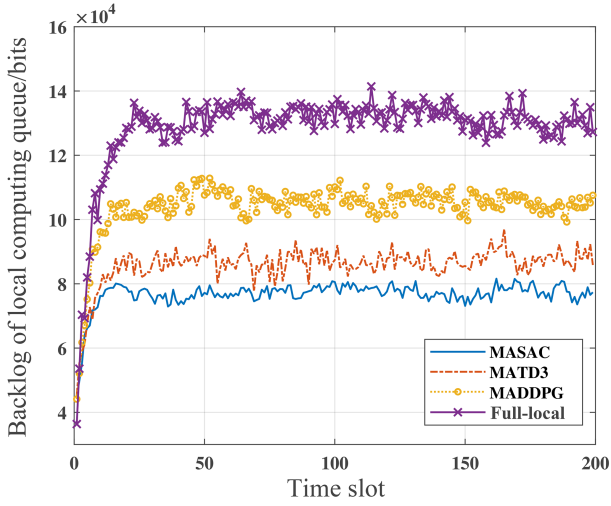


图 5 (网络版彩图) 本地计算队列积压.

Figure 5 (Color online) Local computing queue backlogs.

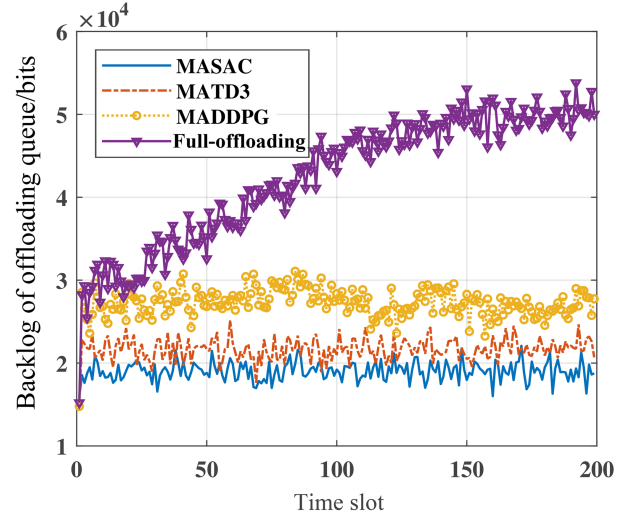


图 6 (网络版彩图) 卸载队列积压.

Figure 6 (Color online) Offloading queue backlogs.

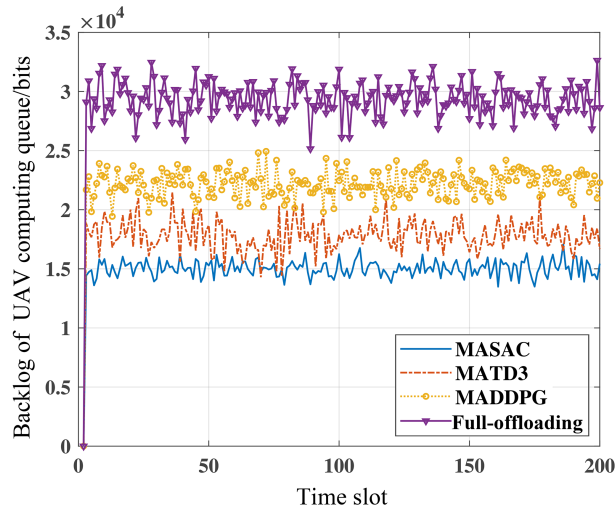


图 7 (网络版彩图) 边缘计算队列积压.

Figure 7 (Color online) Edge computing queue backlogs.

产生较大的卸载和无人机队列积压,不利于系统任务处理的稳定.此外,在所有算法中, MASAC 产生了除全卸载方法外最低的能耗,在第 80 个时隙处所产生的系统平均能耗分别为 MATD3, MADDPG 和全本地计算方案所产生能耗的 68.9%, 45.8% 和 32.2%, 表现出了优异的降低能耗性能.

图 9 和 10 分别显示了虚拟队列 $Z_n^{\text{loc},(H)}(t)$ 和 $Z_n^{\text{loc},(I)}(t)$ 的平均积压情况.其中 MASAC 在本地计算队列的 URLLC 约束中体现了更好的性能,实现了更好的平衡.而全本地计算方法虚拟队列积压值明显高于其他三种方法,这是因为用户设备计算资源有限,无法及时处理产生的全部任务,因违反 URLLC 约束而导致较大的虚拟队列积压.

图 11 展示了本文算法随着李雅普诺夫权重参数 V 的增加,系统平均能耗和时延的变化情况.其中系统平均能耗随 V 值增加而降低,而排队时延反之.这是因为在本文问题 P2 的加权目标函数中, V 与系统能耗相乘作为惩罚项, V 值越大,优化算法就越偏向于牺牲排队时延性能换取系统能耗的降低.因此,图 11 可以指导 V 的灵活选取,满足实际系统的运行需求.

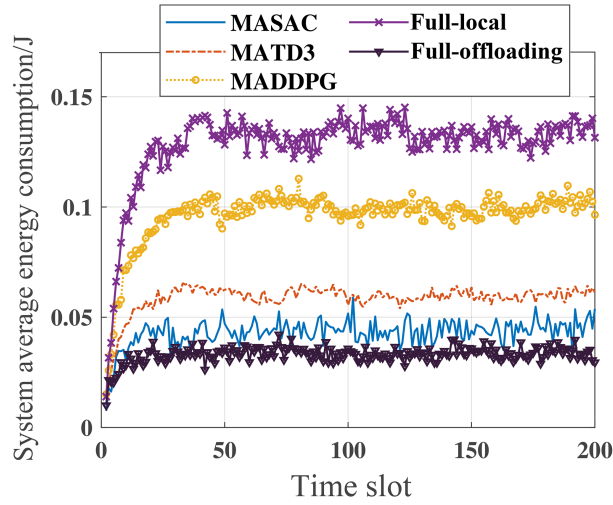


图 8 (网络版彩图) 系统平均能耗.

Figure 8 (Color online) System average energy consumption.

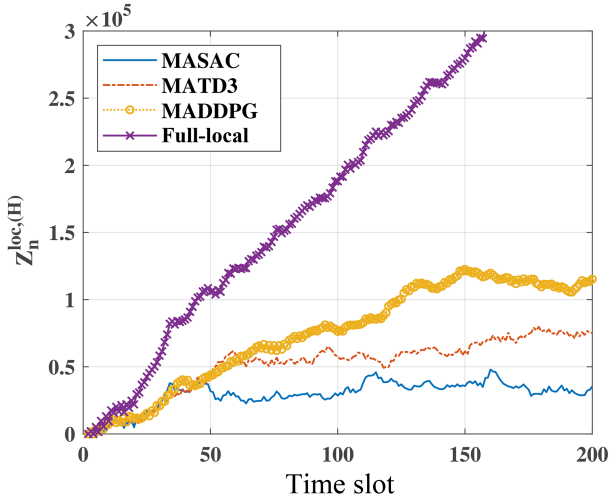
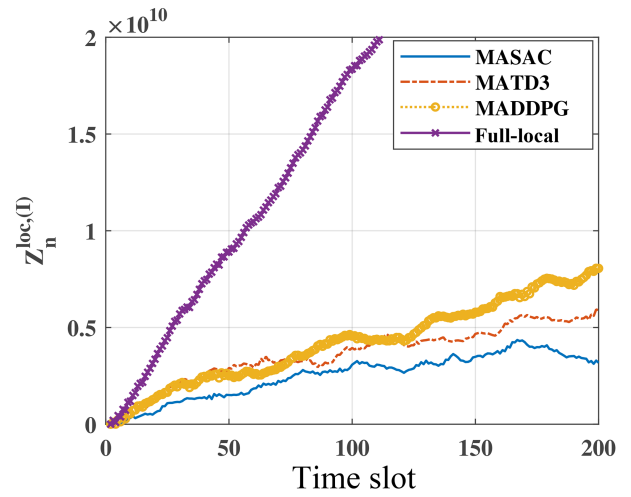
图 9 (网络版彩图) 虚拟队列积压 $Z_n^{\text{loc},(H)}(t)$.Figure 9 (Color online) Virtual queue backlogs of $Z_n^{\text{loc},(H)}(t)$.图 10 (网络版彩图) 虚拟队列积压 $Z_n^{\text{loc},(I)}(t)$.Figure 10 (Color online) Virtual queue backlogs of $Z_n^{\text{loc},(I)}(t)$.

图 12 给出了不同算法得到的 URLLC 可靠性 - 时延曲线, 其中横轴为最大容忍排队时延, 纵轴的可靠性定义为所有时隙中各队列的实际排队时延不大于容忍时延的概率. 可见随着容忍时延的增加, 对队列中积压任务量的限制被放宽, 可靠性呈非递减趋势, 这反映了 URLLC 的可靠性 - 时延基本权衡. 此外, 本文所提 MASAC 的曲线显著高于基准方法, 这是因为本文通过任务卸载、无人机轨迹优化和计算资源分配增强了系统的任务处理性能, 更利于提升可靠性.

图 13 模拟了任务数据量较大 (平均数据量为 [19, 28] Mbits), 且考虑卫星移动性的场景 (在卫星不可见时隙内, 终端只能选择无人机进行任务卸载). 可见本文算法通过合理调配本地和边缘侧的算力资源仍能有效完成任务处理, 由于需要占用更多 CPU 和功率资源, 能耗相对于低数据量场景有所增加. 当卫星不可见时隙数为 10 时, 能耗相比卫星持续可见的情况仅增加了 7.47%, 这说明所提方法能够适应边缘节点的移动性, 实现卫星和无人机之间任务卸载的灵活切换.

图 14 展示了不同系统规模下本文所提算法的平均能耗. 一方面, 随着 UAV 数量的增加, 能耗呈现先下降后趋于平缓的趋势. 这是因为设备更容易选择信道条件优质的 UAV 卸载任务, 同时 UAV 侧可提供更多的边缘计

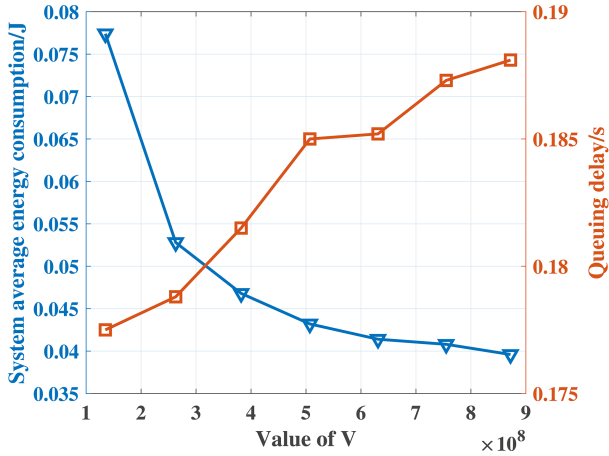
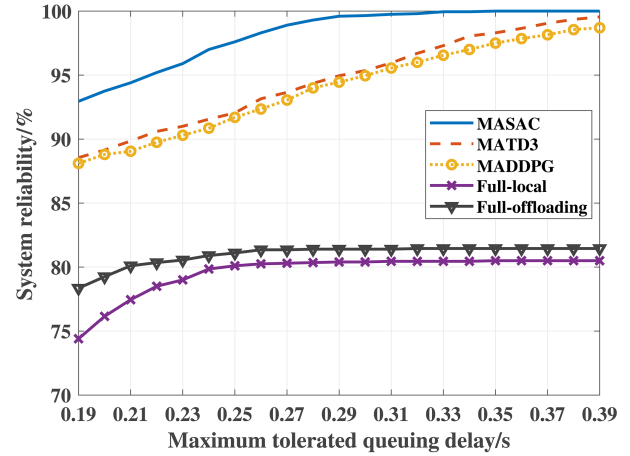
图 11 (网络版彩图) 系统性能随权重参数 V 的变化。Figure 11 (Color online) System performance versus weighting parameter V .

图 12 (网络版彩图) URLLC 可靠性 - 时延曲线。

Figure 12 (Color online) Reliability-delay curves of URLLC.

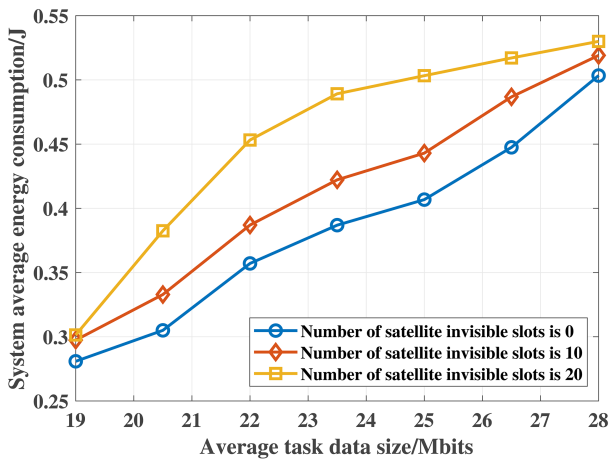


图 13 (网络版彩图) 能耗随平均任务数据量的变化。

Figure 13 (Color online) Energy consumption versus average task data amount.

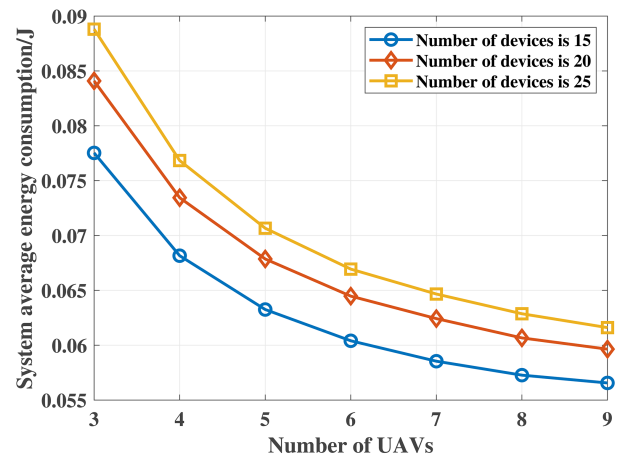


图 14 (网络版彩图) 能耗随 UAV 和设备数量的变化。

Figure 14 (Color online) Energy consumption versus number of UAVs and number of devices.

算资源进行任务处理. 但 UAV 数量的进一步增加会导致防碰撞约束 (9) 愈发难以满足, 从而限制轨迹优化的自由度, 减缓了能耗降低的趋势. 另一方面, 用户设备增多会加剧同频干扰和资源竞争, 因而系统能耗增加. 此外, 该结果证实了本文算法在不同系统规模下都能有效输出优化结果, 具备可扩展性.

6 结论

本文研究了卫星和无人机辅助的空天地协同 MEC 系统中用户任务切分、卸载策略、功率控制、无人机轨迹规划和边缘计算资源分配的联合优化问题, 旨在使终端设备总体能耗最小. 具体而言, 采用李雅普诺夫优化将短期卸载决策、长期任务排队延迟以及 URLLC 约束高度耦合的原始问题解耦为一系列可逐时隙求解的问题. 然后, 本文提出了一种基于 MASAC 的 URLLC 感知能耗优化算法, 在 MASAC 熵最大化、集中训练分散执行等框架优势的基础上, 设计改进的重参数化及激活函数调整策略, 使其能够输出混合动作变量, 从而良好适配本文考虑的空天地协同赋能 MEC 环境. 通过该算法连续求解子问题能够动态平衡 URLLC 约束和能量消耗, 实现

URLLC 感知. 仿真结果验证了所提方法在平衡系统能耗和任务队列稳定性方面具有优越的整体性能.

参考文献

- 1 Qin P, He H T, Zhao X W, et al. Efficient resource allocation with context-awareness for parked car road side unit-based Internet of Vehicles. *J Commun*, 2022, 43: 113–125 [秦鹏, 和昊婷, 赵雄文, 等. 基于停放车辆路边单元环境感知的车联网资源高效分配. *通信学报*, 2022, 43: 113–125]
- 2 Li B, Fei Z, Zhang Y. UAV communications for 5G and beyond: recent advances and future trends. *IEEE Int Things J*, 2019, 6: 2241–2263
- 3 Siddavaatam R, Woungang I, Carvalho G H S, et al. Mobile cloud storage over 5G: a mechanism design approach. *IEEE Syst J*, 2019, 13: 4060–4071
- 4 Li B, Fang T. Application and prospect of Beidou navigation satellite system (BDS) in smart grid. *Electric Power*, 2020, 53: 107–116 [李博, 方彤. 北斗卫星导航系统 (BDS) 在智能电网的应用与展望. *中国电力*, 2020, 53: 107–116]
- 5 Li J S, Yuan X D, Zhang H Y, et al. Concept, architecture, and key technologies of a highly resilient space-air-ground integrated communication and coordination power distribution network. *Automat Electric Power Syst*, 2025, 49: 1–10 [林加顺, 袁晓冬, 张宸宇, 等. 空天地一体化通信协同的高弹性配电网概念、架构与关键技术. *电力系统自动化*, 2025, 49: 1–10]
- 6 Qin P, Wang M, Zhao X W, et al. Content service oriented resource allocation for space-air-ground integrated 6G networks: a three-sided cyclic matching approach. *IEEE Int Things J*, 2022, 10: 828–839
- 7 Zhou Q, Zhou K, Jin Q R, et al. Novel applications of satellite internet enabling new power system. *Automat Electric Power Syst*, 2024, 48: 42–61 [周全, 周柯, 金庆忍, 等. 卫星互联网赋能新型电力系统创新应用. *电力系统自动化*, 2024, 48: 42–61]
- 8 Gao B T, Hu Z Y, Wang W S, et al. Review on fast active power control and frequency support technologies of renewable energy stations. *Proc CSEE*, 2024, 44: 4335–4353 [高丙团, 胡正阳, 王伟胜, 等. 新能源场站快速有功控制及频率支撑技术综述. *中国电机工程学报*, 2024, 44: 4335–4353]
- 9 Li P, Xi W, Yu H, et al. Analysis of key issues and prospect for digital transformation of distribution networks based on edge computing. *Automat Electric Power Syst*, 2024, 48: 29–41 [李鹏, 习伟, 于浩, 等. 基于边缘计算的配电网数字化转型关键问题分析与展望. *电力系统自动化*, 2024, 48: 29–41]
- 10 Li C C, Zhang Y S, Sun C H. Computation offloading for satellite-based edge computing in LEO satellite network. *Radio Commun Tech*, 2022, 48: 401–407 [李诚成, 张亚生, 孙晨华. 低轨卫星互联网星载边缘计算卸载策略. *无线电通信技术*, 2022, 48: 401–407]
- 11 Shen Y, Zhou Q, Wen Y, et al. Integrated satellite-terrestrial network framework for next generation smart grid. *IEEE Trans Smart Grid*, 2024, 15: 5249–5252
- 12 Liu J, Zhao X, Qin P, et al. Learning-based multi-UAV assisted data acquisition and computation for information freshness in WPT enabled space-air-ground PIOT. *IEEE Trans Netw Sci Eng*, 2024, 11: 48–63
- 13 Shen X H, Gu L, Chen Y, et al. Research on UAV-SWIPT optimization method for space-air-ground integrated power Internet of Things based on MADDPG. *Comput Eng*, 2026, doi: 10.19678/j.issn.1000-3428.0252202 [神显豪, 顾玲, 陈谊, 等. 基于 MADDPG 的空天地电力物联网 UAV-SWIPT 优化方法研究. *计算机工程*, 2026, doi: 10.19678/j.issn.1000-3428.0252202]
- 14 Qin P, Fu M, Fu Y, et al. Cooperative UAV trajectory design and resource allocation in blockchain-enabled secure aerial edge computing network. *IEEE Trans Wireless Commun*, 2026, doi: 10.1109/TWC.2025.3582151
- 15 Tun Y K, Kim K T, Zou L, et al. Collaborative computing services at ground, air, and space: an optimization approach. *IEEE Trans Veh Technol*, 2023, 73: 1491–1496
- 16 Zhang P, Li Y, Kumar N, et al. Distributed deep reinforcement learning assisted resource allocation algorithm for space-air-ground integrated networks. *IEEE Trans Netw Serv Manage*, 2022, 20: 3348–3358
- 17 Ma T, Zhou H, Qian B, et al. UAV-LEO integrated backbone: a ubiquitous data collection approach for B5G Internet of remote things networks. *IEEE J Sel Areas Commun*, 2021, 39: 3491–3505
- 18 Abdel-Aziz M K, Samarakoon S, Liu C F, et al. Optimized age of information tail for ultra-reliable low-latency communications in vehicular networks. *IEEE Trans Commun*, 2019, 68: 1911–1924
- 19 Li B, Liu W S, Fei Z S. Partial computation offloading for mobile edge computing in space-air-ground integrated network. *J Electron Inform Tech*, 2022, 44: 3091–3098 [李斌, 刘文帅, 费泽松. 面向空天地异构网络的边缘计算部分任务卸载策略. *电子与信息学报*, 2022, 44: 3091–3098]
- 20 Qin P, Fu Y, Zhao X, et al. Optimal task offloading and resource allocation for C-NOMA heterogeneous air-ground integrated power internet of things networks. *IEEE Trans Wireless Commun*, 2022, 21: 9276–9292
- 21 Wu Q, Zhang R. Common throughput maximization in UAV-enabled OFDMA systems with delay consideration. *IEEE Trans Commun*, 2018, 66: 6614–6627
- 22 Zhou C, Wu W, He H, et al. Deep reinforcement learning for delay-oriented IoT task scheduling in SAGIN. *IEEE Trans Wireless Commun*, 2020, 20: 911–925

- 23 Islam S M R, Avazov N, Dobre O A, et al. Power-domain non-orthogonal multiple access (NOMA) in 5G systems: potentials and challenges. *IEEE Commun Surv Tutor*, 2016, 19: 721–742
- 24 Long H, Xu C, Zheng G, et al. Socially-aware energy-efficient task partial offloading in MEC networks with D2D collaboration. *IEEE Trans Green Commun Netw*, 2022, 6: 1889–1902
- 25 国家电网公司. Q/GDW 677–2011 分布式电源接入配电网监控系统功能规范. 北京: 中国电力出版社, 2011
- 26 Neely M J. *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Synthesis Lectures on Learning, Networks, and Algorithms. Cham: Springer, 2010
- 27 Bao W, Chen H, Li Y H, et al. Joint rate control and power allocation for non-orthogonal multiple access systems. *IEEE J Sel Areas Commun*, 35: 2798–2811
- 28 Qin Y, Zhang Z, Li X, et al. Deep reinforcement learning based resource allocation and trajectory planning in integrated sensing and communications UAV network. *IEEE Trans Wireless Commun*, 2023, 22: 8158–8169
- 29 Qin P, Wang S, Fu M, et al. Air-ground integrated network resource optimization based on MATD3. *Sci Sin Inform*, 2024, 54: 1474–1486 [秦鹏, 王硕, 付民, 等. 基于 MATD3 的空地网络资源优化. *中国科学: 信息科学*, 2024, 54: 1474–1486]
- 30 Chen G, Xing Z W, Shen F, et al. Joint user association and dynamic resource allocation algorithm for LEO-RAN slicing scenarios. *J Commun*, 2024, 45: 173–187 [陈赓, 邢治薇, 沈斐, 等. LEO-RAN 切片场景联合用户关联和动态资源分配算法. *通信学报*, 2024, 45: 173–187]

Space-air-ground integrated network-empowered mobile edge computing: MASAC-based URLLC-aware energy optimization

Peng QIN*, Jinghan LI & Yang FU

School of Electrical and Electronic Engineering, North China Electric Power University, Beijing 102206, China

* Corresponding author. E-mail: qinpeng@ncepu.edu.cn

Abstract To fulfill the flexible, efficient, and random access demands as well as quality of service guarantee requirements of diverse terminals in the new power system, it is imperative to tackle the incomplete communication coverage, weak real-time task response, and device energy limitation issues of the current terrestrial network. In this paper, we propose a space-air-ground integrated network-empowered mobile edge computing system model consisting of low-earth orbit satellites, unmanned aerial vehicles (UAVs), and ground devices. We elaborate on the joint optimization problem of task splitting, task offloading, power control, UAV trajectory planning, and edge computational resource allocation, with the aim of minimizing the long-term terminal energy consumption. However, this problem is intractable due to the coupling between short-term optimization decisions and long-term ultra-reliable and low-latency communication (URLLC) constraints. To this end, we employ Lyapunov optimization to transform the original problem into a series of subproblems that can be solved in a slot-by-slot fashion. We then design a multi-agent soft actor-critic (MASAC)-based URLLC-aware energy optimization algorithm, together with improved reparameterization and activation function configuration tricks, to efficiently solve them. Numerical results verify that the proposed approach outperforms other benchmark methods in reducing system energy consumption whilst guaranteeing URLLC constraints.

Keywords space-air-ground collaborative mobile edge computing, trajectory optimization, resource allocation, ultra-reliable and low-latency communication, multi-agent soft actor-critic (MASAC)