

天地一体化边缘计算网络服务迁移算法研究

冯伊凡^① 吴畏虹^{*①} 孙罡^① 王颖^② 罗龙^① 虞红芳^①

^①(电子科技大学 成都 611731)

^②(紫金山实验室 南京 211111)

摘要: 针对天地一体化边缘计算网络(STECN)的高动态性和复杂性, 如何协同优化用户服务延迟与系统迁移成本成为服务迁移算法设计的关键问题。因此, 该文提出一种多智能体服务迁移优化(MASMO)算法。首先, 考虑到低轨卫星的有限覆盖时间、网络拓扑的动态变化和卫星节点资源等多重因素, 对用户服务延迟和系统迁移成本进行建模。其次, 将服务迁移优化问题进一步建模为多智能体马尔可夫决策过程(MAMDP)。随后, 采用基于轨迹感知的状态信息增强方法, 通过融合卫星轨道的可预测信息, 引导智能体学习具备前瞻性与稳定性的迁移行为。最后, 基于循环多智能体近端策略优化(rMAPPO)算法对服务迁移优化问题进行求解, 以最大程度地降低用户服务延迟和系统长期迁移成本。仿真结果表明, 所提算法具有良好的收敛性, 能够有效协调服务延迟与迁移成本之间的矛盾, 对用户服务延迟降低2.90%~14.63%的同时, 有效降低了系统服务迁移成本10.66%~30.57%。

关键词: 天地一体化网络; 服务迁移; 多智能体深度强化学习

中图分类号: TN92

文献标识码: A

文章编号: 1009-5896(2026)02-0499-13

DOI: 10.11999/JEIT250835

CSTR: 32379.14.JEIT250835

1 引言

当前, 低轨卫星正从传统的通信平台向具备在轨处理能力的智能边缘节点演进^[1]。低轨卫星(Low Earth Orbit satellite, LEO)凭借其广泛覆盖能力, 对广泛接入的计算服务提供了可能性^[2]。随着低轨卫星数量的增加以及卫星板载处理能力的提升, 通过低轨卫星所搭载的计算、存储和网络资源作为边缘节点处理计算任务已经成为相关领域的研究热点。目前一些主要云服务提供商如Google, Microsoft与AWS等, 正通过与航天企业合作, 积极探索将云计算能力部署至轨道平台^[3]。

天地一体化边缘计算网络(Satellite-Terrestrial integrated Edge Computing Networks, STECN)是实现计算能力轨道部署的关键技术架构。其中具备算力、存储和服务调度能力的低轨卫星逐步成为更靠近用户的边缘云节点^[4,5], STECN架构通过跨空间域、跨层级的资源^[6]与计算服务协同, 能够满足多样化应用对低延迟、高可用性、泛在接入的算力服务的需求。

由于低轨卫星的独有特性, 当用户设备从一个边缘节点移动到另一个节点时, 进行服务迁移是确保STECN架构中服务连续性的重要手段^[7,8]。相较

于相对静态、资源稳定的地面网络, 低轨卫星网络具备高度时空动态性、资源有限性等特征。具体而言, 低轨卫星的轨道运行速度较快, 频繁变化的可视窗口和高度动态的拓扑结构^[9], 产生了延迟波动、服务中断等服务可用性问题^[10], 使得静态部署的服务难以持续稳定运行^[11]。服务迁移能够在保障用户服务连续性的同时, 使得提高服务质量成为可能。

然而, STECN的高度时空动态性给服务迁移与部署问题带来了挑战, 一方面迁移决策需考虑空间维度上动态变化的网络拓扑和节点资源状态, 另一方面迁移决策具有时间依赖性, 单步决策不仅影响当前的空间资源分布, 而且影响后续网络状态的演化和服务质量变化。此外, 随着低轨卫星数量的提升和业务空间尺度增长, 服务迁移模式逐步面临迁移成本和服务延迟的双重挑战。从迁移成本的角度看, 服务镜像、运行状态、用户上下文数据等信息的同步与资源的重新调度等操作均会产生显著的迁移成本^[12], 如果涉及多跳星间链路, 长距离的数据转发会造成迁移成本风险进一步加剧。从服务延迟的角度看, 延迟优化是卫星边缘计算的关键研究目标之一^[13], 用户任务处理延迟、网络传输时延和服务迁移延迟等直接决定了业务服务质量^[14]。频繁的迁移容易造成服务资源的不稳定与性能波动, 导致用户体验降低, 甚至是服务中断或不可用风险, 尤其对实时性要求较高的应用而言, 这种影响尤为突出。

近年来已有研究从不同角度优化卫星网络环境中的服务迁移和部署问题。文献[14]针对低轨卫星网络任务卸载过程中的服务迁移问题, 提出一种基

收稿日期: 2025-09-01; 改回日期: 2025-12-19; 网络出版: 2025-12-23

*通信作者: 吴畏虹 wuweihong@uestc.edu.cn

基金项目: 国家自然科学基金(62394324)

Foundation Item: The National Natural Science Foundation of China (62394324)

于对决双深度Q学习(Dueling-Double Deep-Q-Learning, D3QN)算法的分布式方案,以最小化服务时延和优化卸载路径选择。文献[15]提出了一种基于马尔可夫决策过程(Markov Decision Process, MDP)的卫星移动边缘计算服务迁移在线优化模型,以实现服务延迟的优化,并利用分布式两层分解方法,降低了问题的计算复杂度。文献[16]考虑基于网络功能虚拟化的卫星服务部署,提出了一种两阶段联合优化框架,以最小化任务时延与设备能耗。上述研究大多将卫星网络建模为准静态拓扑,部署决策主要依赖当前状态,忽略未来可达性变化容易导致迁移策略的短视性,使系统在长时间尺度上产生冗余迁移,难以实现性能的长期最大化。一些研究通过引入动态决策机制来解决该问题。文献[17]通过可重构的时间扩展图来解决空地一体化网络的动态性和复杂性,并进一步提出了动态服务功能链调度模型。文献[10]考虑到空地一体化网络的动态性和不稳定性,设计了面向资源故障的服务恢复模型以解决服务中断问题。为适应LEO卫星的高度动态场景,文献[18]基于人工电场算法提出一种服务迁移和计算卸载协同驱动的动态决策机制。文献[19]提出了一种面向卫星边缘计算架构的在线服务部署算法,通过融合Lyapunov优化理论与Gibbs sampling技术,旨在最大化系统的长期平均服务覆盖率。上述研究中主要针对优化服务延迟、服务覆盖率、设备能耗等指标展开,缺少对于低轨卫星网络综合成本的考虑。

不同于地面网络,低轨卫星的运动轨迹基于星历数据是确定的,其具有可预测性的可视窗口、邻接关系^[20]为服务迁移策略中引入动态的网络状态信息提供了可能。因此,本文以最大程度地减少用户服务延迟和系统迁移成本为目标,提出了一种基于多智能体强化学习的服务迁移算法:

首先,基于STECN架构对服务迁移问题进行建模,并纳入低轨卫星动态性、节点资源约束、链路间歇性连通等影响用户服务延迟和迁移成本的实际因素。

随后,将上述优化问题建模为多智能体马尔可夫决策过程,并提出一种多智能体服务迁移优化(Multi-Agent Service Migration Optimization, MASMO)算法。基于星历数据的确定性和卫星网络的可预测性,提出基于轨迹感知的状态信息增强方法。采用循环多智能体近端策略优化(recurrent Multi-Agent Proximal Policy Optimization, rMAPPO)算法,利用循环神经网络的时序特征提取能力,捕获服务迁移的时间依赖性,训练智能体

学习具有长期规划能力的迁移策略,实现用户服务延迟与系统迁移成本的联合优化。

通过在STECN环境下进行仿真验证,对所提算法进行性能评估。结果表明,MASMO算法能够协调服务延迟与迁移成本之间的矛盾,在优化用户服务延迟的同时,有效降低了系统的服务迁移成本。

2 服务迁移模型

服务迁移模型同步考虑用户服务延迟与服务迁移成本进行迁移策略的优化。服务迁移优化框架所包含的模型如下:

首先,对STECN的服务迁移场景进行建模,对LEO卫星网络进行模型化定义;

随后,对服务迁移需求进行建模以表述服务迁移的事件与需求;

其次,分别对服务迁移成本与用户服务延迟进行建模,对框架模型中的优化目标进行表述;

最后,以上述模型为基础建立优化问题。

2.1 STECN服务迁移场景模型

本文旨在研究面向STECN的服务迁移问题,首先对STECN服务迁移场景进行建模。如图1所示,考虑一个搭载边缘计算服务器的低轨卫星网络,其中 $S = \{s_1, s_2, \dots, s_N\}$ 表示由 N 颗LEO卫星构成的节点集合, $U = \{u_1, u_2, \dots, u_M\}$ 表示由 M 个地面用户构成的集合,可根据自身任务需求向卫星边缘服务器发起服务请求。服务以虚拟机或容器实例的形式部署于卫星节点的边缘服务器上,设服务类型集合为 $E = \{e_1, e_2, \dots, e_I\}$,其中每种服务类型 $e_i \in E$ 可在一个或多个卫星节点上以服务实例的形式运行。

当用户 $u \in U$ 处于某颗卫星 $s \in S$ 的可视范围内时,用户可通过该卫星访问其边缘服务器上部署的服务实例。随着卫星的轨道运动导致用户与原服务节点之间的延迟不满足服务要求,或原节点因资源受限无法继续响应服务请求,为保证服务的连续性、降低时延并提升系统整体性能,系统需根据当前网络状态和节点资源情况制定服务迁移策略,将已部署的服务实例迁移至其他候选卫星节点。

为建模和分析卫星网络中随时间变化的拓扑结构与服务动态迁移过程,本文将连续时间划分为等长离散时隙,记为 $\mathcal{T} = \{1, 2, \dots, T_{\max}\}$ 。在每个时隙内,满足可视关系的卫星之间通过星间链路(Inter-Satellite Link, ISL)建立通信连接,每颗卫星最多与四颗相邻卫星建立连接,LEO卫星网络的连接关系可以用连通性图 $G(t) = (S(t), \mathcal{E}(t))$,其中 $\mathcal{E}(t)$ 表示有效星间链路集合, $(s_i, s_j) \in \mathcal{E}(t)$ 表示卫星 s_i 与 s_j 在时隙 t 存在有效通信链路。

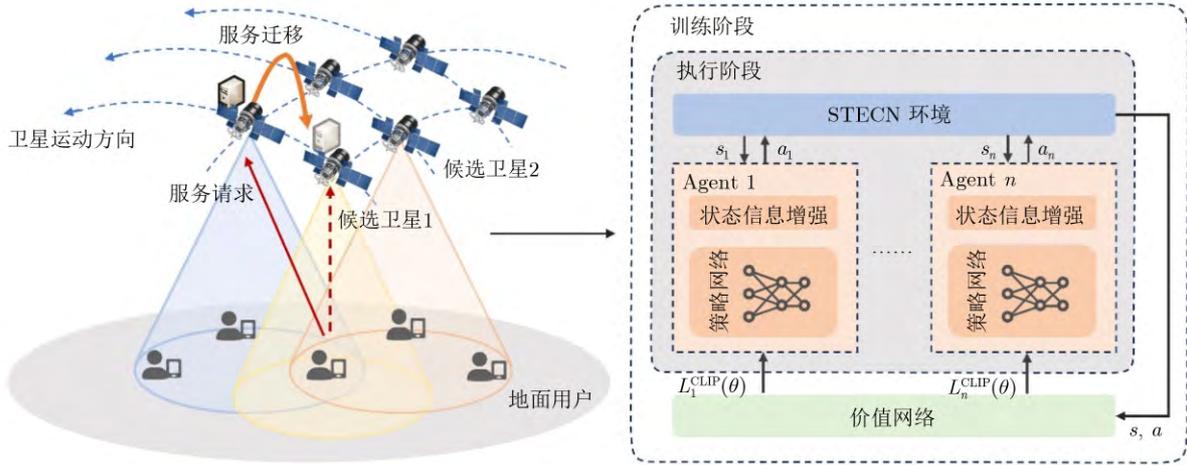


图1 天地一体化边缘计算网络服务迁移场景

2.2 服务迁移需求模型

在时隙 $t \in \mathcal{T}$, 每个用户 $u \in U$ 可能发起计算服务请求, 该请求需调用某类特定服务实体 $e_i \in E(t)$. 服务实体 e 表示一类可被多个用户共享的计算服务, 定义为 $e = \{I_e, \tau_e\}$, 其中 I_e 表示服务镜像数据大小, τ_e 表示其最大可容忍延迟. 服务实体 e 部署在至少一个卫星节点 $s \in S$ 上, 其部署状态用变量 $x_{e,s}(t) \in (0, 1)$ 表示, 若值为1表示服务实体 e 在 t 时刻部署于节点 s 上. 二元变量 $y_{u,e}(t) \in (0, 1)$ 表示用户请求的服务类型, 若值为1表示用户 u 请求服务实体 e . 服务实体可在不同时隙间在卫星节点间迁移, 服务迁移动作通过二元变量 $m_{i,j}^e(t)$ 表示, 若在 t 时刻服务实体 e 从节点 i 迁移至 j , 则 $m_{i,j}^e(t) = 1$. 服务迁移路径需依据动态变化的网络拓扑确定. 在每个时隙 t , 系统可基于当前的连通性图 $G(t) = (S(t), \mathcal{E}(t))$ 计算从源节点到目标节点的最短路径.

2.3 网络模型

本文考虑的网络由LEO卫星星座与地面用户组成. 由于LEO卫星的位置动态变化, 只有当LEO卫星与地面用户满足特定几何关系时二者才能进行通信. 卫星 s_i 在时隙 t 的位置为 $\mathbf{p}_i(t)$, 地面用户 u 的位置由经纬度 (ϕ_u, λ_u) 给出. 卫星与用户之间的可见性取决于用户对卫星的仰角是否超过阈值 θ_{\min} [21]. 仰角计算公式为

$$\theta_{u,i}(t) = \arctan \left(\frac{h_{u,i}(t)}{\sqrt{l_{u,i}^2(t) - h_{u,i}^2(t)}} \right) \quad (1)$$

其中, $l_{u,i}(t)$ 表示用户与卫星的欧氏距离, $h_{u,i}(t)$ 表示相对高度差. 若 $\theta_{u,i}(t) \geq \theta_{\min}$, 则认为卫星 s_i 对用户 u 可见.

卫星间的链路连接随卫星在轨道上的位置不断变化, 记 $\mathbf{p}_i(t)$ 表示卫星 s_i 在时隙 t 的三维位置向量, 则任意两颗卫星 s_i 与 s_j 的几何距离为

$$l_{i,j}(t) = \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|_2 \quad (2)$$

根据距离可定义星间链路是否存在的可见性函数表示为

$$\chi_{i,j}(t) = \begin{cases} 1, & l_{i,j}(t) \leq D_{\max} \\ 0, & \text{其他} \end{cases} \quad (3)$$

其中 D_{\max} 为最大可建立链路的距离阈值.

在满足可见性的基础上, 本文采用+grid方式 [18] 构建星间拓扑, 即每颗卫星与轨道内前后相邻卫星及轨道间相邻卫星建立链路, 形成卫星间的网格结构拓扑. 综上, 可以得到时隙 t 下的网络拓扑

$$G(t) = (S \cup U, \mathcal{E}_{\text{sat}}(t) \cup \mathcal{E}_{\text{user}}(t)) \quad (4)$$

其中, $\mathcal{E}_{\text{sat}}(t)$ 为卫星间链路集合, 由所有满足 $\chi_{i,j}(t) = 1$ 的卫星对构成, $\mathcal{E}_{\text{user}}(t)$ 为满足可见性条件的星地链路集合.

2.4 服务延迟模型

用户的服务延迟由多个部分组成, 包括用户与卫星节点之间的通信延迟, 节点处理延迟, 以及服务迁移延迟.

用户设备与卫星节点之间的通信延迟通常包含用户数据的传输延迟和网络传播延迟. 不失一般性, 计算任务回传数据远小于用户上传数据, 因此只考虑上行链路的通信延迟 [14], 可以表示为

$$d_{\text{comm}}^{u,i}(t) = \frac{D_u}{R_{u,i}(t)} + \frac{l_{u,i}}{c} \quad (5)$$

其中, D_u 表示用户上行链路的传输数据大小, $l_{u,i}$ 表示在 t 时刻用户 u 与卫星 i 之间的距离, c 表示光速. 另外 $R_{u,i}(t)$ 表示用户 u 与可见卫星 i 之间的信道容量, 基于Shannon理论计算得到

$$R_{u,i} = B_{u,i} \cdot \log_2 \left(1 + \frac{P_u h_{u,i}}{N_0 B_{u,i}} \right) \quad (6)$$

其中, $B_{u,i}$ 表示链路带宽, P_u 表示发射功率, $h_{u,i}$

表示信道增益, N_0 表示噪声功率频谱密度。信道增益 $h_{u,i}$ 由发射端与接收端天线增益及自由空间路径损耗决定

$$h_{u,i}(t) = \frac{G_t G_r \lambda^2}{(4\pi d_{u,i}(t))^2} \quad (7)$$

其中, G_t, G_r 分别为发射端与接收端天线增益, λ 为载波波长。

为满足服务的正常运行, 边缘节点为每个服务分配一定的计算资源。卫星节点 i 的计算资源分配表示为 $f_i = \{f_i^{e_1}, f_i^{e_2}, \dots, f_i^{e_k}\}$, 其中 k 为节点 i 上部署的服务实体数量。假设 t 时刻由卫星节点 i 处理用户 u 的服务请求, 则计算延迟可以表示为

$$d_{\text{comp}}^{u,i} = \frac{\omega_e r_{u,e}}{f_i^{e_i}} \quad (8)$$

若服务 e 在 t 时刻迁移至节点 i , 迁移延迟包括源服务器的停止延迟, 计算服务器之间的同步延迟以及目标服务器的启动延迟, 因此可以表示为

$$d_{\text{mig}}^{e,\text{src},i} = d_{\text{stop}}^i + \sum_p r_{u,e} \left(\frac{I_e}{R_{p_n,i}} + \frac{l_{p_n,i}}{c} \right) + d_{\text{start}}^i \quad (9)$$

其中, p 表示源节点到节点 i 的路径, 由最短路算法得到; $R_{i,j}$ 表示该星间链路的传输速率; $l_{i,j}$ 表示节点间的距离。因此, 用户 u 的总体服务延迟由传输延迟、计算延迟和迁移延迟相加得到

$$d_{\text{total}}^{u,i} = d_{\text{comm}}^{u,i} + d_{\text{comp}}^{u,i} + d_{\text{mig}}^{e,\text{src},i} \cdot m_{\text{src},i}^e \quad (10)$$

2.5 服务迁移成本模型

当服务实体从卫星节点 i 迁移到 j 时, 将引发额外的系统开销, 即迁移成本。迁移成本可以包含多种成本因素, 例如数据传输所消耗的链路资源、服务中断时间等^[22]。根据现有研究^[23,24], 服务迁移成本可表示为迁移开销 $b(x)$ 和数据传输开销 $c(y)$ 两部分。 $b(x)$ 主要刻画服务迁移过程中不可避免的服务中断时间, 其中 $x = \|s_i - s_j\|$ 表示节点 i 和 j 之间距离。服务迁移所消耗的通信资源与服务中断时间通常随迁移距离的增加而上升, 因此可以将迁移成本建模为与节点 i 和 j 之间距离相关的非递减函数。此外, 服务迁移后, 当前用户需传输任务状态信息到目标节点, 使其继续执行任务, 因此会产生与用户距离相关的数据传输成本 $b(y)$, 其中 $y = \|u - s_j\|$ 表示目标节点 j 与用户 u 之间的距离。所以服务迁移成本可以表示为

$$w_e^{s_i, s_j}(t) = b(\|s_i - s_j\|) + c(\|u - s_j\|) \quad (11)$$

2.6 问题建立

基于上述卫星动态性模型对 STECN 场景进行建模, 本文旨在优化该动态场景下卫星的服务迁移

决策。结合对服务延迟模型和迁移成本模型的分析, 本文的研究目标是在满足用户服务请求、系统资源等约束条件下, 最小化用户服务延迟与长期迁移成本组成的系统综合成本。该优化问题可以描述为

$$\begin{aligned} \min \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T \sum_{u \in U} \omega_1 x_{e,s_i}(t) y_{u,e}(t) d^{u,i} + \omega_2 w_e(t) \quad (12) \\ \text{s.t. C1: } \sum_{s \in s_u(t)} x_{e,s}(t) \geq y_{u,e}(t), \forall u, \forall e, \forall t \\ \text{C2: } x_{e,s}(t) \cdot y_{u,e}(t) = 0, s \notin S_u, \forall u, e, s, t \\ \text{C3: } \sum_{e \in E} x_{e,s}(t) \cdot c_e \leq C_s, \forall s, t \\ \text{C4: } \sum_{e \in E} x_{e,s}(t) \cdot m_e \leq M_s, \forall s, t \\ \text{C5: } x_{e,s}(t) \cdot y_{u,e}(t) \cdot d^{u,e}(t) \leq D_e^{\text{max}}, \forall u, e, s, t \end{aligned} \quad (13)$$

其中, 约束 C1 表示保证每个服务请求在任一时隙均有明确的卫星节点进行服务部署; 约束 C2 表示当服务发生迁移时, 迁移的目标节点必须处于用户的可见范围内; 约束 C3~C4 表示卫星节点所承载的全部服务所需的存储和计算资源总和不得超过节点自身的资源容量; 约束 C5 表示服务请求的总延迟必须满足服务类型规定的最大容忍延迟。

3 多智能体服务迁移优化算法

式(12)所示的问题是一个连续优化问题, 且具有高度复杂性和非凸性, 难以直接求解。为解决该问题, 本文利用其满足马尔可夫决策过程(Markov Decision Process, MDP)对问题进行转换。首先将服务迁移问题建模为一个多智能体马尔可夫决策过程(Multi-Agent Markov Decision Process, MAMDP), 并在集中训练、分布执行(Centralized Training with Decentralized Execution, CTDE)的框架下, 提出了一种基于循环多智能体近端策略优化算法(rMAPPO)的服务迁移优化算法。通过结合基于轨迹感知的状态信息增强方法(Trajectory-Aware State Enhancement, TASE)改善状态表征, 使MASMO算法能够在动态环境中实现对服务延迟与迁移成本的联合优化。

3.1 多智能体马尔可夫决策过程

在服务迁移过程中存在多个分布式卫星节点的协作行为, 同时服务迁移决策需要在动态环境中根据本地有限观测信息做出动作选择。本研究将上述多卫星协作的服务迁移问题建模为一个MAMDP, 将卫星作为代理, 并用一个关键元组 $\langle N, O, A, P, R, \gamma \rangle$

来定义该问题。其中, N 表示智能体数量, 每个卫星节点作为一个智能体; O 表示全局状态空间, 由所有智能体的观测信息组成, 记为 $O_t = \{o_t^i | i = 1, 2, \dots, N\}$; A 表示联合动作空间, $A_t = \{a_t^i | i = 1, 2, \dots, N\}$ 表示 t 时刻的动作空间, 包含每个智能体在每个时隙的可选迁移动作; 状态转移概率 $P = (o_t | o_{t-1}, a_t)$ 表示执行动作后环境状态由 o_{t-1} 变为 o_t 的转移概率; R 表示奖励函数; $\gamma \in [0, 1)$ 表示折扣因子。在强化学习框架下, 智能体的目标是最大化长期累积奖励 $\sum_{t=0}^T \gamma^t R_t$ 。因此, 根据优化目标, 我们将单步奖励 R_t 设计为服务延迟与迁移成本的负加权和, 以鼓励智能体学习降低系统成本的策略。为了确保决策始终满足约束C1~C4, 在策略执行过程中采用了动作屏蔽机制, 智能体只能在满足硬约束的有效动作集合中进行选择。为满足服务延迟约束C5, 我们在奖励函数中引入了惩罚项, 当任意用户的服务延迟超过其服务类型规定的最大容忍延迟时, 系统将获得一个额外的惩罚, 激励智能体提高系统的服务可用性。

(1)状态空间: 本文中服务迁移问题的状态空间由每个智能体的状态空间得到。由于每个卫星智能体只能获取自身环境中的局部观测信息, 并据此选择动作 a_t^i , 其在每个时隙 t 的状态空间 o_t^i 定义为

$$o_t^i = \{U_{\text{visible},t}, T_t, L_t^j, U_{\text{service},t}, R_t^i\} \quad (14)$$

其中包括当前卫星的可见用户列表 $U_{\text{visible},t}$, 用户请求的任务数据大小 T_t , 以及与目标卫星 j 之间的距离以及链路带宽 L_t^j , 卫星当前服务的用户列表 $U_{\text{service},t}$, 以及当前可用的剩余计算资源和存储资源 R_t^i 。

(2)动作空间: 动作空间的定义体现了服务迁移的两重选择, 即“是否迁移”以及“迁移到何处”。每个智能体根据当前的局部状态 o_t^n , 基于策略 π_θ^n 执行的动作可以表示为一个2元组

$$a_t^n = \langle m_t^n, d_t^n \rangle \quad (15)$$

其中 $m_t^n \in \{0, 1\}$ 表示是否执行迁移操作; $d_t^n \in D_t^n$ 表示迁移目标节点编号, D_t^n 表示满足约束C2的所有可迁移目标卫星。动作空间的规模随卫星节点数和可服务实体数量而变化, 假设每颗卫星上有 S 个可迁移服务实体, 每个服务的可迁移目标节点数为 M^s , 则智能体的动作空间维度为 $(\sum_s^S M^s + 1)$ 。

(3)状态转移: 环境状态的转移机制由两部分共同决定, 一是轨道动力学模型, 使卫星在时隙间发生位置与可见性关系的变化; 二是智能体动作的选择, 影响服务驻留的分布与系统资源占用情况。

因此, 系统状态在 $t \rightarrow t+1$ 的演化过程可表示为 $o_{t+1} = f(o_t, a_t)$ 。

(4)奖励函数: 在奖励函数设计方面, 本研究以同时最小化用户平均服务延迟与长期迁移成本为目标, 因此在每个时隙 t , 智能体 n 的即时奖励 r_t^n 定义为

$$r_t^n = \omega_1 \left(L_{\text{base}} - \frac{1}{U} \sum_{u=1}^U x_{e,s_i}(t) y_{u,e}(t) d^{u,i} \right) - \omega_2 W \quad (16)$$

其中, L_{base} 为参考延迟基线值, ω_1, ω_2 分别为延迟奖励权重和系统迁移开销惩罚权重。为实现全局策略优化, 在时隙 t 的全局奖励为每个智能体的局部奖励之和, 表示为

$$r_t = \sum_{n=1}^N r_t^n \quad (17)$$

3.2 基于轨迹感知的状态表示增强

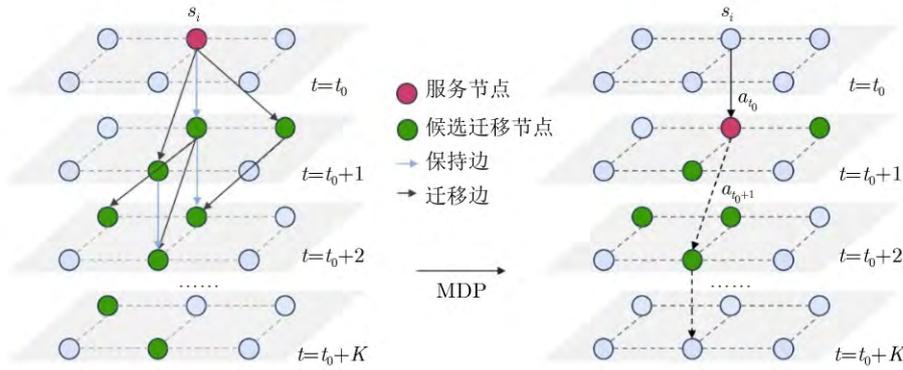
在STECN中, 服务迁移优化不仅与单步决策相关, 还受到未来若干时隙拓扑演化的影响。为捕捉STECN高度动态的拓扑变化, 本文提出一种基于轨迹感知的状态表示增强方法(TASE), 利用卫星轨迹的可预测性, 构建以当前服务节点为中心的局部时间扩展图(Time-Expanded Graph, TEG), 将物理拓扑的动态演进映射为图结构的时序分析。随后通过一种序列化策略将图结构映射为高维时序特征向量, 并利用循环神经网络(Recurrent Neural Network, RNN)提取隐含窗口内拓扑变化信息的状态特征。从而增强智能体对环境状态的感知, 使其能够感知未来的拓扑演化趋势, 学习具有长期规划能力的迁移策略。

TASE采用滑动窗口机制进行状态更新, 在每个时隙 t 智能体基于星历数据和局部观测, 构建覆盖时间窗口 $[t, t+K]$ 的局部有向无环图 $G_{i,t:t+K}$ 。考虑到全局TEG规模庞大导致状态空间爆炸, 采用以当前服务卫星 s_i 为中心的剪枝策略构建局部TEG, 保留当前服务卫星及其邻居节点。随着时间推进至 $t+1$, 预测窗口同步向后滑动, TEG被重新构建以适应新的网络状态。定义 $G_{i,t:t+K} = (\mathcal{V}_{\text{TE}}, \mathcal{E}_{\text{TE}})$ 为以卫星 s_i 为中心、预测窗口长度为 K 的有向无环图, 如图2所示, 其中:

(1)节点集 \mathcal{V}_{TE} : \mathcal{V}_{TE} 包含未来 K 个时隙内, 卫星 s_i 及其邻居卫星集合 \mathcal{N}_i 中相关的时空节点副本, 表示为

$$\mathcal{V}_{\text{TE}} = \{v_{j,\tau} | j \in \{i\} \cup \mathcal{N}_i, \tau \in \{t, t+1, \dots, t+K\}\} \quad (18)$$

(2)时序边集 \mathcal{E}_{TE} : 边集 \mathcal{E}_{TE} 描述了网络状态在时间维度的演化, 包含两类对应于MDP中动作空

图2 以卫星 s_i 为中心的局部TEG及动态MDP

间的有向边。其中，保持边表示服务继续驻留，若卫星 s_i 在相邻时隙均满足可见性约束，则存在边 $(v_{i,\tau}, v_{i,\tau+1})$ 。迁移边用于表示可迁移动作，当卫星 s_i, s_j 之间存在ISL且卫星 s_j 满足约束C2~C4时，建立边 $(v_{i,\tau}, v_{j,\tau+1})$ 。边的权重 $w(e)$ 赋值为对应动作的迁移成本，由式(11)得到。

为了将非欧几里得结构的 $G_{i,t:t+k}$ 输入至深度神经网络，本文将TEG中的拓扑信息转化为多通道时序序列。利用LEO卫星星座在空间分布上呈现出的规则网格状拓扑特性，对于任意服务节点，将其邻居集合映射至固定的语义方位通道 $\mathcal{N}_{\text{dir}} = \{\text{Left, Right, Front, Back}\}$ 。定义时刻 τ 的局部状态向量 \mathbf{x}_τ 为

$$\mathbf{x}_\tau = [f_{\text{hold}}^{(\tau)}, \mathbf{f}_{\text{mig},d}^{(\tau)}], d \in \mathcal{N}_{\text{dir}} \quad (19)$$

其中， $f_{\text{hold}}^{(\tau)}$ 表示保持边的属性特征， $\mathbf{f}_{\text{mig},d}^{(\tau)}$ 表示向固定方位 d 邻居迁移的特征向量，定义为

$$\mathbf{f}_{\text{mig},d}^{(\tau)} = [\mathbb{I}_d, C_{\text{mig}}, C_{\text{out}}^{\text{min}}, D_{\text{out}}] \quad (20)$$

其中 $\mathbb{I}_d \in (0, 1)$ 用于表示边的存在性，若边不存在，则特征向量被掩码处理。 C_{mig} 表示该节点所对应的迁移成本， $C_{\text{out}}^{\text{min}} = \min\{w(e) | e \in \mathcal{E}_{\text{out}}(v_{d,\tau})\}$ 表示从该节点出发的最小迁移成本， D_{out} 为出度，反映该节点的拓扑鲁棒性。

因此整个预测窗口内的TEG被转化为一个有序的时间序列矩阵 $\mathbf{X}_{\text{TE}} = [\mathbf{x}_t, \mathbf{x}_{t+1}, \dots, \mathbf{x}_{t+K}]^T$ 。为了捕捉序列 \mathbf{X}_{TE} 中隐含的拓扑演化信息和长期依赖关系，构建基于门控循环单元(Gate Recurrent Unit, GRU)的编码器。将时刻 τ 的局部状态向量 \mathbf{x}_τ 输入GRU，经过 K 个时间步的递归计算，即

$$\mathbf{h}_\tau = \text{GRU}(\mathbf{x}_\tau, \mathbf{h}_{\tau-1}) \quad (21)$$

其中， $\mathbf{h}_{\tau-1}$ 为上一时刻的隐状态。最终，取隐状态 \mathbf{h}_{t+K} 作为编码器的输出 \mathbf{h}_{TE} ， \mathbf{h}_{TE} 作为局部TEG的状态特征，与智能体当前局部观测联合构成扩展状态输入，以显式引入未来拓扑演化特征。

3.3 MASMO算法

为解决STECN场景下高维状态空间与动态特性带来的决策难题，本文在MAPPO算法的基础上，提出一种融合轨迹感知增强的多智能体服务迁移优化算法(MASMO)。本算法采用CTDE框架，引入TASE方法增强智能体对时变环境的表征能力，并结合神经网络增强了对时序相关性的建模能力。在集中训练阶段，价值网络接收全局状态输入，输出联合状态值函数，用于提升优势函数估计的准确性；在分布执行阶段，每个智能体仅依赖局部观测与预测特征进行决策，保证算法的分布式可执行性。

(1)状态特征融合：针对时刻 t 的实时观测状态 o_t ，输入多层感知机(Multi-Layer Perceptron, MLP)得到实时状态向量 \mathbf{h}_{obs} 。为了让策略网络能够同时利用实时状态及局部TEG状态信息，将 \mathbf{h}_{obs} 和 \mathbf{h}_{TE} 在特征维度上进行拼接，构成联合状态向量作为策略网络和价值网络的输入

$$\mathbf{o}_{\text{joint}} = [\mathbf{h}_{\text{obs}} \oplus \mathbf{h}_{\text{TE}}] \quad (22)$$

(2)策略网络(Actor)：策略网络采用带有循环单元的神经网络结构，以每个智能体的局部观测信息作为输入，输出该智能体在当前状态下的动作分布 $\pi_\theta(a_i | \mathbf{o}_{\text{joint},i})$ 。策略网络的优化目标为最大化所有智能体的期望累积奖励，其目标函数采用裁剪的PPO形式可以表示为

$$L(\theta) = \mathbb{E}_t [\min(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1 - \varepsilon, 1 + \varepsilon)A_t) + \beta H(\pi_\theta)] \quad (23)$$

其中， $r_t(\theta)$ 为新旧策略比， A_t 为优势函数， $H(\pi_\theta)$ 为策略的熵项。

(3)价值网络(Critic)：价值网络用于评估全局状态下各智能体联合动作的长期回报。价值网络参数记为 ϕ ，其输入为所有卫星智能体的联合观测 s_t ，输出状态值函数 $V_\phi(s_t)$ 。价值网络的优化目标是 minimize 时序差分误差，损失函数定义为

$$L(\phi) = \mathbb{E} \left[(V_\phi(s_t) - R_t)^2 \right] \quad (24)$$

其中 R_t 为实际累积回报。Critic的存在能够为Actor提供更准确的优势估计，从而提升策略优化的稳定性。

(4)算法训练流程：在轨迹采样阶段，所有卫星智能体与环境并行交互，每个智能体在时隙 t 基于局部观测与预测增强特征生成状态输入。策略网络将输入映射为动作分布 $\pi_\theta(a_i | \mathbf{o}_{\text{joint},i}, \mathbf{h}_{t-1})$ ，其中 \mathbf{h}_{t-1} 为上一时刻的隐状态。智能体根据该分布采样动作 $a_i(t)$ ，决定是否迁移及目标卫星编号。执行动作后，环境返回新的观测 $\mathbf{o}_{\text{joint},i}(t+1)$ 、全局状态 s_{t+1} 、即时奖励 $r(t)$ 等信息。这个过程会持续进行，直到一个完整的回合结束或达到预设的最大步长，从而为每个智能体生成一条经验轨迹。这些轨迹数据将被存储在一个经验回放池中，用于后续的网络参数更新。

在收集到足够多的经验数据后，训练过程将进入网络更新阶段。该阶段会从经验回放池中随机抽取一个批次(mini-batch)的数据，并对Actor和Critic网络进行多轮的迭代更新。首先，更新Critic网络，目标是准确地估计在全局状态 s_t 下的状态价值。为了降低价值估计的方差并平衡偏差，采用广义优势估计(Generalized Advantage Estimation, GAE)方法来计算优势函数 \hat{A}_t

$$\hat{A}_t = \sum_{l=0}^{T-t} (\gamma \lambda)^l (r_{t+l} + \gamma V_\phi(s_{t+l+1}) - V_\phi(s_{t+l})) \quad (25)$$

其中 $V_\phi(s)$ 为价值网络估计的状态值。Critic网络的损失函数 $L(\phi)$ 定义为其预测的状态价值与目标价值之间的均方误差，通过最小化该损失函数，采用梯度下降法对Critic网络的参数 ϕ 进行更新。其次，更新Actor网络 π_θ ，其更新目标是最大化PPO的裁剪替代目标函数 $L(\theta)$ ，旨在限制单次更新的步长，从而保证训练的稳定性。 \hat{A}_t 是由Critic网络计算出的优势函数， ε 是一个用于限制裁剪范围的超参数。为了鼓励探索，通常会在目标函数中加入一个熵奖励项 $S[\pi_\theta]$ 。因此，最终的Actor网络损失函数表示为

$$J(\theta) = L(\theta) + c \cdot S[\pi_\theta](s_t) \quad (26)$$

其中 c 是熵项的系数。通过对该目标函数进行梯度上升，即可更新行动者网络的参数 θ 。由于网络是循环结构，梯度将通过时间反向传播(BackPropagation Through Time, BPTT)进行计算。综上所述，MASMO算法的整体训练流程如算法1中所示。

(4)复杂度分析：MASMO算法分为集中训练和分布执行两个阶段，其中算法的主要计算开销在

训练阶段，其计算任务是更新所有智能体的Actor网络和中心的Critic网络。Actor网络以维度为 D_o 的局部观测向量为输入，输出维度与动作空间大小 D_a 成正比，因此复杂度约为 $O(L_A \cdot W_A^2 + W_A \cdot (D_o + D_a))$ ，其中 L_A 和 W_A 分别为网络层数与平均宽度。Critic网络需要处理由 N 个智能体观测拼接形成的全局状态，因此其复杂度约为 $O(L_C \cdot W_C^2 + W_C \cdot N \cdot D_o)$ ，其中 L_C 和 W_C 分别为网络层数与平均宽度。由于Critic的输入维度随智能体数量线性增长，训练阶段总体复杂度对 N 呈线性依赖。在执行阶段，由于各卫星独立并行决策，复杂度仅与智能体自身神经网络的规模和局部观测的维度有关。

4 仿真结果与分析

4.1 实验设置

本文的仿真实验在一台配备了2个NVIDIA RTX 3090 GPU, 128 GB内存的服务器上进行，软件环境为TensorFlow 2.0.0和Python 3.10.13。为构建低轨卫星网络拓扑，本文采用Skyfield库对卫星的轨道运动进行建模。实验场景设定为由4颗轨道高度为212 km、轨道倾角为51.67°的LEO卫星组

算法1 MASMO算法

输入：环境状态 O_t
输出：最优服务迁移策略 π_θ

- (1) 初始化actor网络 π_θ 与critic网络 V_ϕ ，初始化经验回放缓冲区 D 。
- (2) for 训练迭代次数 $k = 1, 2, \dots, K$ do
- (3) 清空经验回放缓冲区 D 。
- (4) 重置并行环境 $e = 1, 2, \dots, N_{\text{env}}$
- (5) for 并行环境 $e = 1, 2, \dots, N_{\text{env}}$ do
- (6) for $t = 1, 2, \dots, T$ do
- (7) 基于局部观测与预测增强特征构造状态 $\mathbf{o}_{\text{joint},t}$ 。
- (8) 通过执行当前策略 π_θ 与环境交互，采集一条经验轨迹 τ_e 。
- (9) 将轨迹 τ_e 存入经验回放缓冲区 D 。
- (10) end for
- (11) 保存当前策略参数 $\theta_{\text{old}} \leftarrow \theta$ 。
- (12) 利用critic网络 V_ϕ 和采集到的数据计算优势估计和 \hat{A} 回报目标 \hat{R} 。
- (13) for 更新轮次 $u=1,2,\dots,U$ do
- (14) 从 D 中随机抽取 n 个经验作为一个 mini-batch b ;
- (15) 对于 mini-batch b 中的每个数据块 c do
- (16) 使用数据块首帧的隐藏状态更新 π 和 V 的RNN状态。
- (17) 通过最小化损失函数 $L(\phi)$ 更新critic参数 ϕ 。
- (18) 通过最大化目标函数 $J(\theta)$ 更新actor参数 θ 。
- (19) end for
- (20) end for

成的网络,轨道参数参考Starlink星座。在实验周期设置方面,单个实验周期 $T=600\text{ s}$,每个时隙 $t=20\text{ s}$ 。实验过程中,用户请求在周期内至少需要经历1次服务迁移,以验证算法在高度动态的STECN网络中的性能表现。地面用户随机分布在给定区域范围内,其服务请求服从泊松分布。本文提出的MASMO算法的策略网络结构为双层全连接神经网络,Actor网络与Critic网络均包含两个隐藏层,每层包含64个神经元。具体参数设置如表1所示。为验证所提MASMO算法在不同场景下的性能,本文将其与MAPPO, MADDPG, Greedy及Random 4种基准算法进行对比:

(1)多智能体深度确定性策略梯度(MADDPG)算法:是一种基于Actor-Critic架构的离策略(Off-policy)算法,它将基于确定性策略梯度(DDPG)扩展至多智能体环境,在集中训练分布执行(CTDE)

框架下通过经验回放机制学习确定性策略,利用集中式Critic评估联合动作以缓解多智能体环境的非平稳性。

(2)贪婪最近节点(Greedy):一种贪婪算法,在每个决策时隙,系统总是将服务迁移到距离用户瞬时位置最近的可见卫星节点,以追求当前最低的通信延迟。

(3)随机策略(Random):在每个决策时隙,从所有可用的可见卫星节点中随机选择一个作为服务节点,作为性能基准的下限。

4.2 仿真结果分析

图3展示了对MASMO算法的性能分析。其中图3(a)展示了不同学习率(lr)设置下的收敛表现,可以观察到随着训练步数的增加,lr=5e-4, lr=5e-5以及lr=5e-6 3条曲线均表现出收敛趋势,验证了本文所提算法可行性与有效性。在训练初期,lr=5e-4的收敛速度最快,但由于在梯度迭代过程中步长过大,在100 000步后出现了较明显的震荡,导致模型容易在非凸优化空间中错失全局最优解,从而陷入次优解。lr=5e-5和lr=5e-6曲线的初始学习速度较慢,但避免了过早收敛,获得了更高的奖励值,在高噪声的多智能体环境中表现出更强的稳定性。其中lr=5e-5收敛速度和奖励值表现最优,因此后续实验的学习率均设置为5e-5。此外,为分析折扣因子的选择对算法性能的影响,图3(b)展示了当 γ 分别取0.90, 0.95和0.99时,算法的平均回合奖励收敛曲线。当 γ 设置为0.95时,算法取得了最优的性能,其平均回合奖励收敛至约19.5,并且收敛过程稳定。在高度动态的卫星网络环境中,取值适中、略偏向于远视的折扣因子能够最好地平衡即时奖励与长期回报。从图3(c)中可以看出, MADDPG算法收敛速度较慢,奖励值在1 000轮次时收敛至12左右。MASMO算法与MAPPO算法的收敛速度相当,但相较于MAPPO算法, MASMO算法通过TASE方法能够将奖励函数的收敛值提高11.47%。

图3(d)展示了奖励权重变化对算法性能的影响。设置延迟奖励权重 ω_1 从0.1变化至0.9,同时设置迁移成本权重 $\omega_2 = 1 - \omega_1$ 。随着 ω_1 的增加,算法对优化用户服务延迟的优先级提高,服务被迁移到更靠近用户的卫星上,从而带来迁移次数的增长。当 ω_1 持续增大时,优化服务延迟的收益逐渐受限,为平衡服务延迟和迁移成本,后续实验中 ω_1 取值为0.6。

图4展示了用户数量从4增加到16时,不同算法在平均用户服务延迟、平均迁移成本以及延迟构成上的性能表现。从图4(a)可以观察到,随着用户数

表1 参数设置

| 参数 | 值 |
|--------------------------|----------------|
| 卫星轨道高度 h (km) | 212 |
| 卫星轨道面倾角($^\circ$) | 51.67 |
| 卫星节点所配备的计算资源 (Gcycles/s) | [1 000, 2 000] |
| 卫星节点所配备的存储资源(GB) | [2, 3] |
| 地面用户坐标经度范围($^\circ$) | [30, 50] |
| 地面用户坐标纬度范围($^\circ$) | [100, 135] |
| 最小仰角($^\circ$) | 30 |
| 地面用户请求任务大小(kB) | [200, 500] |
| 服务实体大小 I_e (MB) | [300, 500] |
| 任务请求所需要的计算资源 (Gcycle/s) | [100, 200] |
| 任务请求所需要的计算强度 (cycle/bit) | 10^6 |
| 地面用户设备的发射功率 P_u (W) | 5 |
| 地面用户到卫星之间的信道带宽 (MHz) | 8 |
| 地面用户设备天线发射增益 (dBi) | 5 |
| 卫星天线发射增益 (dBi) | 20 |
| 卫星天线接收增益 (dBi) | 40 |
| 实验场景周期 T (s) | 600 |
| 单个时隙 t (s) | 20 |
| K | 4 |
| ω_1, ω_2 | 0.6, 0.4 |
| 训练轮次 | 1800 |
| 经验回放缓冲区大小 | 10^5 |
| 学习率lr | 5e-5 |
| 折扣因子 γ | 0.95 |
| λ | 0.95 |
| mini-batch大小 | 80 |

量的增加，5种算法的平均用户延迟均呈上升趋势。这是由于用户规模扩大后，卫星计算资源竞争加剧，链路占用率提升，导致整体服务时延增加。在小规模用户场景下，Greedy短期延迟表现较优。但随着用户数量增加，该策略忽略了服务迁移成本和长期资源竞争效应，导致延迟增长幅度加快。基于学习的3种算法(MASMO, MAPPO, MADDPG)通过集中式训练获得更优的决策能力，整体性能明

显优于Greedy和Random。MADDPG的确定性策略在高度动态的网络场景下泛化能力受限，因此服务延迟增幅波动。本文所提的MASMO算法在所有用户规模下均表现出最低的服务延迟。当用户数量为16时，MASMO算法的平均服务延迟约为105.03 ms，相比于MAPPO, MADDPG, Greedy和Random算法分别降低了2.90%, 6.78%, 11.01%和14.63%。

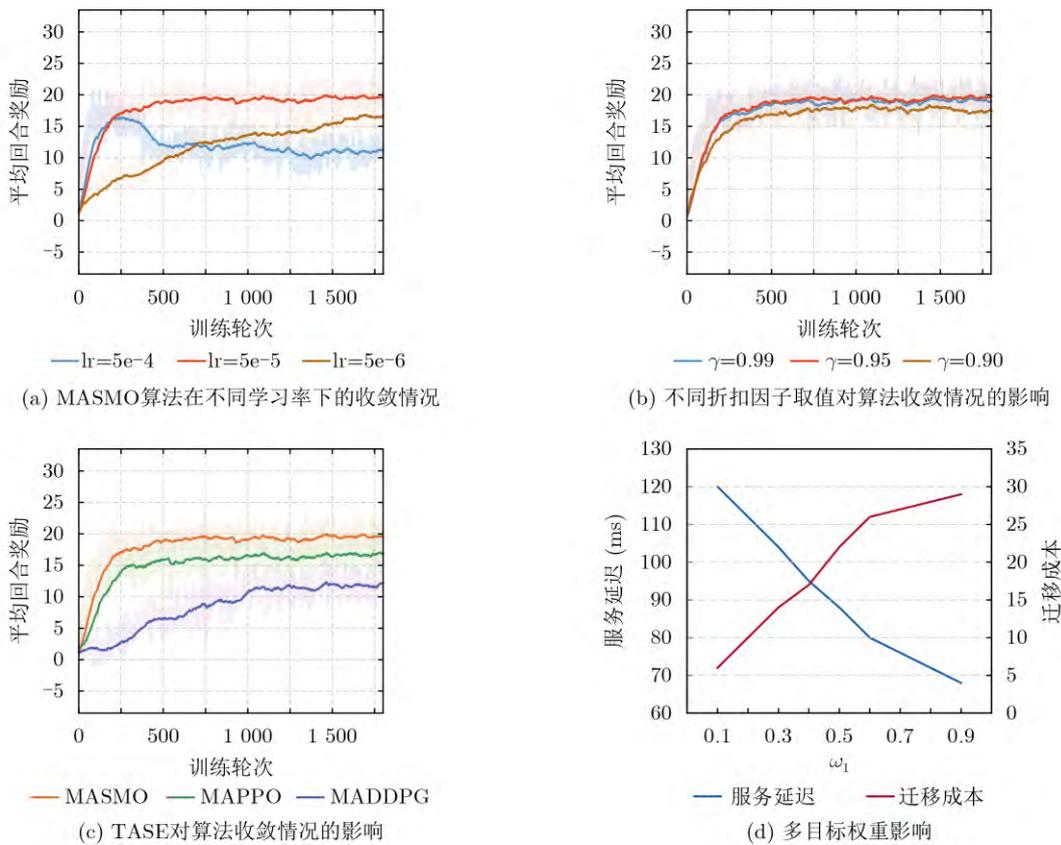


图3 MASMO算法的性能分析

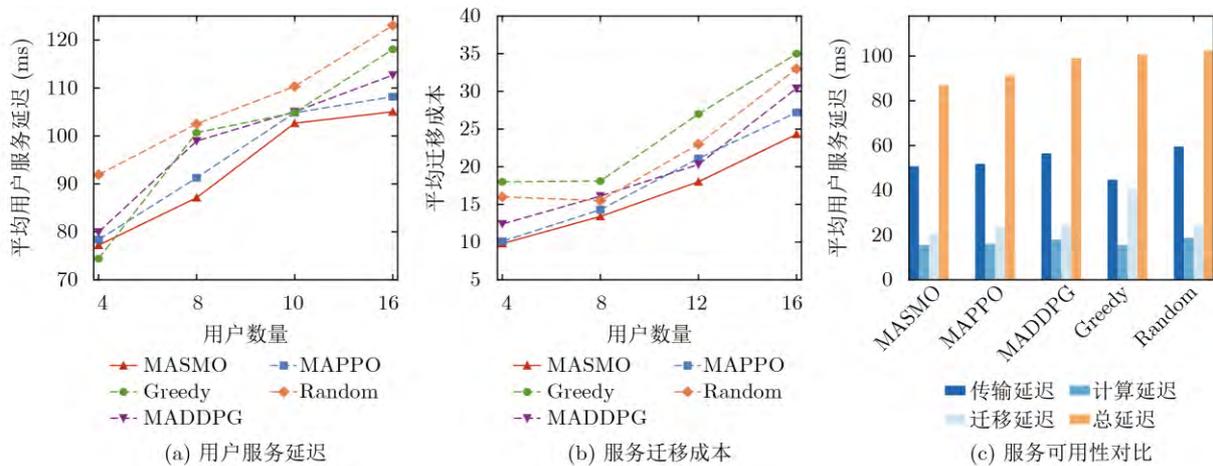


图4 用户数量对算法性能的影响

图4(b)展示了不同算法下的平均迁移成本随用户数量的变化趋势。随着用户数量增加,由于更多用户接入带来了更高的服务请求动态性和更密集的服务切换,所有算法的迁移成本均有所上升。在基线算法中, Greedy与Random由于缺乏对迁移开销的建模,在所有场景下成本始终较高。MADDPG和MAPPO均将迁移成本纳入奖励设计,但对动态拓扑的适应性不足。MASMO算法的迁移成本曲线增长较为平缓,其在追求低延迟的同时,也兼顾了系统迁移成本开销。当用户数量为16时, MASMO的系统迁移成本相较于MAPPO, MADDPG, Greedy和Random分别降低了10.66%, 20.07%, 30.57%和26.36%。为了进一步探究MASMO算法的性能优势,图4(c)给出了用户数量为8时,各算法的延迟组成成分分析。从中可以看出,传输延迟在用户服务延迟中占主导作用。其中, MASMO算法通过对传输延迟和迁移延迟的权衡与优化,实现平均用户服务延迟最低。

图5展示了卫星数量对算法性能的影响。如5(a)所示,从整体趋势来看,随着卫星数量的增加,更多卫星节点可参与服务覆盖,为用户提供了更丰富的可见接入选择,增加了服务资源的冗余性和覆盖率,四种算法的用户延迟均呈下降趋势。随着卫星数量增加, Greedy由于过于追求瞬时最优,而造成服务迁移延迟累积,在8~10颗卫星场景下反而出现延迟反弹。本文所提出的MASMO算法始终保持最低的用户服务延迟。当卫星数量为10时, MASMO的平均服务延迟为70.12 ms,相较于MAPPO算法降低了7.53%,说明MASMO能够实现长时间尺度上更优的迁移决策。5(b)展示了卫星数量对系统平均迁移成本的影响。如图所示, Greedy和Random算法的成本呈现上升趋势, MADDPG算法的成本

存在波动,而MASMO的迁移成本随着卫星数量的增加而稳步降低至10左右。MASMO算法结合前瞻性环境状态感知,在更多的卫星资源场景下做出更优的服务迁移决策,从而减少了因后续网络状态变化而引起频繁服务迁移,因此成本持续降低。

为进一步评估算法的可靠性,引入“任务成功率”和“服务可用性”两个指标,分别表示用户请求的任务在规定的最大容忍延迟内被处理的比例,以及用户服务保持有效连接的时间比例。如图5(c)所示,本文所提的MASMO算法在两个可靠性指标上均取得了最佳性能,任务成功率达到约97.2%,服务可用性达到约95%。这证明了MASMO算法不仅能通过策略学习选出低延迟的服务节点,也能够有效减少迁移次数,从而减少服务中断。

5 结束语

本文研究了STECN场景下的服务迁移问题,针对高动态、资源受限的LEO卫星网络环境,对用户服务延迟和系统迁移成本进行建模,构建了以最小化服务延迟和迁移成本为目标的服务迁移优化问题。为求解最优的服务迁移策略,进一步将该问题建模为多智能体马尔可夫决策过程。其次,针对频繁变化的网络拓扑,通过引入轨迹感知状态增强模块(TASE),结合卫星轨道的可预测性对状态信息进行增强,从而提升策略的前瞻性与稳定性。最后,基于rMAPPO算法,提出了一种多智能体服务迁移优化算法。通过设计联合优化服务延迟与迁移成本的奖励函数,引导智能体学习长期效益最优的迁移策略。仿真结果表明,在不同的卫星数量、用户规模配置下,所提出的MASMO算法在平均服务延迟与迁移成本控制方面优于MAPPO, MADDPG, Greedy以及Random等对比算法。

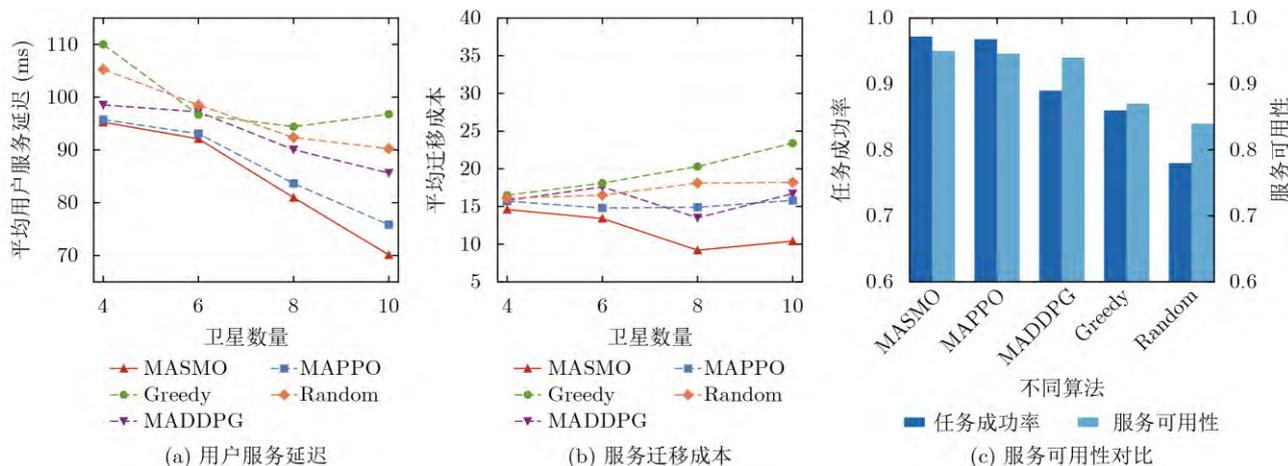


图5 卫星数量对算法性能的影响

参考文献

- [1] 高媛, 方海, 赵扬, 等. 基于自然梯度Actor-Critic强化学习的卫星边缘网络服务功能链部署方法[J]. 电子与信息学报, 2023, 45(2): 455–463. doi: 10.11999/JEIT211384.
- GAO Yuan, FANG Hai, ZHAO Yang, *et al.* A satellite edge network service function chain deployment method based on natural gradient Actor-Critic reinforcement learning[J]. *Journal of Electronics & Information Technology*, 2023, 45(2): 455–463. doi: 10.11999/JEIT211384.
- [2] JIA Min, ZHANG Liang, WU Jian, *et al.* Joint computing and communication resource allocation for edge computing towards Huge LEO networks[J]. *China Communications*, 2022, 19(8): 73–84. doi: 10.23919/JCC.2022.08.006.
- [3] WANG Shangguang and LI Qing. Satellite computing: Vision and challenges[J]. *IEEE Internet of Things Journal*, 2023, 10(24): 22514–22529. doi: 10.1109/JIOT.2023.3303346.
- [4] 王鹏, 张佳鑫, 张兴, 等. 低轨卫星智能多接入边缘计算网络: 需求、架构、机遇与挑战[J]. 移动通信, 2021, 45(5): 35–46. doi: 10.3969/j.issn.1006-1010.2021.05.007.
- WANG Peng, ZHANG Jiaxin, ZHANG Xing, *et al.* Low earth orbit satellite intelligent multi-access edge computing networks: Requirements, architecture, opportunities and challenges[J]. *Mobile Communication*, 2021, 45(5): 35–46. doi: 10.3969/J.ISSN.1006-1010.2021.05.007. doi: 10.3969/j.issn.1006-1010.2021.05.007.
- [5] ZHOU Jian, YANG Qi, ZHAO Lu, *et al.* Mobility-aware computation offloading in satellite edge computing networks[J]. *IEEE Transactions on Mobile Computing*, 2024, 23(10): 9135–9149. doi: 10.1109/TMC.2024.3359759.
- [6] 曹怡璐, 贾子晔, 尤嘉豪, 等. 基于SDN和NFV的空天地一体化网络任务部署与恢复综述[J]. 电信科学, 2025, 41(5): 1–16. doi: 10.11959/j.issn.1000-0801.2025138.
- CAO Yilu, JIA Ziyue, YOU Jiahao, *et al.* A survey of task deployment and recovery in space-air-ground integrated networks based on SDN and NFV[J]. *Telecommunications Science*, 2025, 41(5): 1–16. doi: 10.11959/j.issn.1000-0801.2025138.
- [7] XIE Renchao, TANG Qinqin, WANG Qiuning, *et al.* Satellite-terrestrial integrated edge computing networks: Architecture, challenges, and open issues[J]. *IEEE Network*, 2020, 34(3): 224–231. doi: 10.1109/MNET.011.1900369.
- [8] DENG Peng, GONG Xiangyang, and QUE Xirong. A bandwidth-aware service migration method in LEO satellite edge computing network[J]. *Computer Communications*, 2023, 200: 104–112. doi: 10.1016/j.comcom.2023.01.007.
- [9] HE Lijun, JIA Ziyue, GUO Kun, *et al.* Online joint data offloading and power control for space-air-ground integrated networks[J]. *IEEE Transactions on Wireless Communications*, 2024, 23(12): 18126–18141. doi: 10.1109/TWC.2024.3462349.
- [10] JIA Ziyue, CAO Yilu, HE Lijun, *et al.* NFV-enabled service recovery in space-air-ground integrated networks: A matching game-based approach[J]. *IEEE Transactions on Network Science and Engineering*, 2025, 12(3): 1732–1744. doi: 10.1109/TNSE.2025.3538614.
- [11] WANG Houpeng, GAO Yu'e, GUO Zhonglin, *et al.* Dynamic service migration mechanism in satellite edge computing with location privacy protection[C]. 2024 IEEE 24th International Conference on Communication Technology (ICCT), Chengdu, China, 2024: 1073–1080. doi: 10.1109/ICCT62411.2024.10946657.
- [12] LI Ziqi, ZHANG Heli, LIU Chunyu, *et al.* Online service deployment on mega-LEO satellite constellations for end-to-end delay optimization[J]. *IEEE Transactions on Network Science and Engineering*, 2024, 11(1): 1214–1226. doi: 10.1109/TNSE.2023.3321644.
- [13] KSENTINI A, TALEB T, and CHEN Min. A Markov Decision Process-based service migration procedure for follow me cloud[C]. 2014 IEEE International Conference on Communications (ICC), Sydney, Australia, 2014: 1350–1354. doi: 10.1109/ICC.2014.6883509.
- [14] WU Haonan, YANG Xiumei, and BU Zhiyong. Task offloading with service migration for satellite edge computing: A deep reinforcement learning approach[J]. *IEEE Access*, 2024, 12: 25844–25856. doi: 10.1109/ACCESS.2024.3367128.
- [15] LI Zhen, JIANG Chunxiao, and LU Jianhua. Distributed service migration in satellite mobile edge computing[C]. 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 2021: 1–6. doi: 10.1109/GLOBECOM46510.2021.9685350.
- [16] SUN Jiayu, WANG Huiqiang, NIE Lili, *et al.* A joint strategy for service deployment and task offloading in satellite-terrestrial IoT[J]. *Computer Networks*, 2023, 225: 109656. doi: 10.1016/j.comnet.2023.109656.
- [17] JIA Ziyue, CAO Yilu, HE Lijun, *et al.* Service function chain dynamic scheduling in space-air-ground integrated networks[J]. *IEEE Transactions on Vehicular Technology*, 2025, 74(7): 11235–11248. doi: 10.1109/TVT.2025.3543259.
- [18] WANG Xu, JU Xiaojie, XIE Renchao, *et al.* Service continuity guarantee for coordinated optimization of offloading and migration in LEO satellite computing power networks[C]. 2024 10th International Conference on Computer and Communications (ICCC), Chengdu, China, 2024: 1857–1862. doi: 10.1109/ICCC62609.2024.10942057.
- [19] LI Qing, WANG Shangguang, MA Xiao, *et al.* Service

- coverage for satellite edge computing[J]. *IEEE Internet of Things Journal*, 2022, 9(1): 695–705. doi: 10.1109/JIOT.2021.3085129.
- [20] YU Kangjia, CUI Qimei, LYU Xinchun, *et al.* Efficient collaborative computing for multilayer LEO satellites with spatiotemporal dynamics: A long-term continuous timescale optimization[J]. *IEEE Internet of Things Journal*, 2025, 12(6): 7459–7471. doi: 10.1109/JIOT.2024.3498322.
- [21] BHATTACHERJEE D and SINGLA A. Network topology design at 27, 000 km/hour[C]. The 15th International Conference on Emerging Networking Experiments and Technologies, Orlando, Florida, 2019: 341–354. doi: 10.1145/3359989.3365407.
- [22] WANG Shiqiang, URGAONKAR R, HE Ting, *et al.* Dynamic service placement for mobile micro-clouds with predicted future costs[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2017, 28(4): 1002–1016. doi: 10.1109/TPDS.2016.2604814.
- [23] WANG Shiqiang, URGAONKAR R, ZAFER M, *et al.* Dynamic service migration in mobile edge computing based on Markov decision process[J]. *IEEE/ACM Transactions on Networking*, 2019, 27(3): 1272–1288. doi: 10.1109/TNET.2019.2916577.
- [24] YUAN Quan, LI Jinglin, ZHOU Haibo, *et al.* A joint service migration and mobility optimization approach for vehicular edge computing[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(8): 9041–9052. doi: 10.1109/TVT.2020.2999617.
- 冯伊凡: 女, 硕士生, 研究方向为天地一体化网络。
 吴畏虹: 男, 副研究员, 研究方向为新一代网络。
 孙 罡: 男, 教授, 研究方向为网络虚拟化、区块链技术、人工智能和网络系统安全。
 王 颖: 女, 博士后, 研究方向为卫星网络。
 罗 龙: 女, 副教授, 研究方向为算力网络, 智算网络资源调度。
 虞红芳: 女, 教授, 研究方向为智慧网络及应用研究。

责任编辑: 余 蓉

Service Migration Algorithm for Satellite-terrestrial Edge Computing Networks

FENG Yifan^① WU Weihong^① SUN Gang^① WANG Ying^②
 LUO Long^① YU Hongfang^①

^①(University of Electronic Science and Technology of China, Chengdu 611731, China)

^②(Purple Mountain Laboratories, Nanjing 211111, China)

Abstract:

Objective In highly dynamic Satellite-Terrestrial Edge Computing Networks (STECN), achieving coordinated optimization between user service latency and system migration cost is a central challenge in service migration algorithm design. Existing approaches often fail to maintain stable performance in such environments. To address this, a Multi-Agent Service Migration Optimization (MASMO) algorithm based on multi-agent deep reinforcement learning is proposed to provide an intelligent and forward-looking solution for dynamic service management in STECN.

Methods The service migration optimization problem is formulated as a Multi-Agent Markov Decision Process (MAMDP), which offers a framework for sequential decision-making under uncertainty. The environment represents the spatiotemporal characteristics of a Low Earth Orbit (LEO) satellite network, where satellite movement and satellite-user visibility define time-varying service availability. Service latency is expressed as the sum of transmission delay and computation delay. Migration cost is modeled as a function of migration distance between satellite nodes to discourage frequent or long-range migrations. A Trajectory-Aware State Enhancement (TASE) method is proposed to incorporate predictable orbital information of LEO satellites into the agent state representation, improving proactive and stable migration actions. Optimization is performed using the recurrent Multi-Agent Proximal Policy Optimization (rMAPPO) algorithm, which is suitable for cooperative multi-agent tasks. The reward function balances the objectives by penalizing high migration cost and rewarding low service latency.

Results and Discussions Simulations are conducted in dynamic STECN scenarios to compare MASMO with MAPPO, MADDPG, Greedy, and Random strategies. The results consistently confirm the effectiveness of

MASMO. As the number of users increases, MASMO shows slower performance degradation. With 16 users, it reduces average service latency by 2.90%, 6.78%, 11.01%, and 14.63% compared with MAPPO, MADDPG, Greedy, and Random. It also maintains high cost efficiency, lowering migration cost by up to 30.57% at 16 users (Fig. 4). When satellite resources increase, MASMO consistently leverages the added availability to reduce both latency and migration cost, whereas myopic strategies such as Greedy do not exhibit similar improvements. With 10 satellites, MASMO achieves the lowest service latency and outperforms the next-best method by 7.53% (Fig. 5). These findings show that MASMO achieves an effective balance between transmission latency and migration latency through its forward-looking decision policy.

Conclusions This study addresses the service migration challenge in STECN through the MASMO algorithm, which integrates the TASE method with rMAPPO. The method improves service latency and reduces migration cost at the same time, demonstrating strong performance advantages. The trajectory-enhanced state representation improves foresight and stability of migration behavior in predictable dynamic environments. This study assumes ideal real-time state perception, and future work should evaluate communication delays and partial observability, as well as investigate scalability in larger satellite constellations with heterogeneous user demands.

Key words: Satellite-Terrestrial Edge Computing Network (STECN); Service migration; Multi-Agent Reinforcement Learning (MARL)