

基于多时间尺度协同的无蜂窝 RAN 切片资源分配算法

夏玮玮¹, 王博业¹, 夏雅星¹, 缪巍巍², 汪大洋², 景栋盛³, 燕锋¹, 沈连丰¹

(1. 东南大学移动通信全国重点实验室, 江苏 南京 211102; 2. 国网江苏省电力有限公司信息通信分公司, 江苏 南京 210024;
3. 国网苏州供电公司, 江苏 苏州 215031)

摘要: 针对 6G 无蜂窝无线接入网切片资源分配在业务动态变化时难以保障用户服务质量的问题, 提出了一种基于多时间尺度协同优化的资源分配算法。首先, 将大时间尺度的面向切片的资源配置问题构建为基于均方误差最小化且以用户平均时延为约束的资源需求预测模型, 并通过长短期记忆时序预测网络实现切片资源的精准配置。其次, 将小时间尺度的面向用户的资源分配问题构建为最大化系统效用, 同时保障用户传输速率服务质量指标的资源分配模型。最后, 通过引入基于多智能体协作的近端策略优化算法进行资源的动态分配与实时调整。仿真结果表明, 所提算法能够实现切片资源需求的准确预测, 有效提高系统传输速率, 降低用户平均时延和业务阻塞率。

关键词: 无蜂窝; 网络切片; 资源分配; 多时间尺度; 多智能体强化学习

中图分类号: TN92

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2025121

Cellular-free RAN slicing resource allocation algorithm based on multi-timescale collaboration

XIA Weiwei¹, WANG Boye¹, XIA Yaxing¹, MIAO Weiwei², WANG Dayang²,
JING Dongsheng³, YAN Feng¹, SHEN Lianfeng¹

1. National Mobile Communications Research Laboratory, Southeast University, Nanjing 211102, China
2. Information & Telecommunication Branch, State Grid Jiangsu Electric Power Co., Ltd., Nanjing 210024, China
3. State Grid Suzhou Power Supply Company, Suzhou 215031, China

Abstract: To address the issue of ensuring user quality of service in resource allocation for 6G cell-free radio access network slicing under dynamic service variations, a resource allocation algorithm based on multi-timescale collaborative optimization was proposed. Firstly, the large time-scale slice-oriented resource allocation problem was formulated as a mean square error-minimization-based resource demand prediction model with average user latency constraints, and the accurate allocation of slice resources was achieved by a long and short-term memory time-series prediction network. Secondly, the small time-scale user-oriented resource allocation problem was formulated as a resource allocation model that maximized the system utility while guaranteeing the user transmission rate quality of service metric. Finally, a Proximal allocation algorithm based on Policy optimization multi-agent collaboration was introduced to enable dynamic resource allocation and real-time adjustment. Simulation results demonstrate that the proposed algorithm can accurately predict slice resource, improve the system transmission rates while reducing the average user latency and service blocking probability.

Keywords: cell-free, network slicing, resource allocation, multi-timescale, multi-agent reinforcement learning

收稿日期: 2025-04-17; 修回日期: 2025-06-23

通信作者: 夏玮玮, wwxia@seu.edu.cn

基金项目: 国家电网总部科技项目“超五代无线接入理论研究”(No.5700-202499320A-1-3-ZB)

Foundation Item: The Science and Technology Project of State Grid Corporation of China “Research on Beyond-5G Wireless Access Theory” (No.5700-202499320A-1-3-ZB)

0 引言

物联网 (IoT, Internet of things)、人工智能、工业互联网等领域新业务的蓬勃发展,对网络通信能力提出了更高要求,6G技术具有超高速率、超高可靠性和超低时延等显著特征,高度吻合这些新业务对网络通信能力的需求,将成为终端数据传输的首选通信技术^[1]。传统的蜂窝网络架构存在覆盖范围有限、边缘用户体验不佳以及小区间干扰等问题,而无蜂窝网络架构摒弃了传统蜂窝的概念^[2],通过大量部署分布式接入点 (AP, access point) 实现了更加密集的系统覆盖^[3]和更高的系统吞吐量,有效解决了边缘用户服务质量 (QoS, quality of service) 较低的问题,已成为6G移动通信系统发展的关键架构^[4-5]。

作为6G移动通信的关键技术之一,网络切片技术将物理网络虚拟化为多个逻辑切片,根据特定业务需求量身定制资源分配策略,从而为多类型业务的服务质量需求提供了强有力的保障^[6-7]。目前对于核心网络切片的研究主要聚焦于切片部署和虚拟网络功能 (VNF, virtualized network function) 的布置与管理^[8-9]。无线接入网 (RAN, radio access network) 切片需要兼顾无线信道状态的随机性、无线资源的有限性和RAN架构的多样性,其资源分配更为复杂,成为当前的研究热点^[10]。因此,本文聚焦于研究无蜂窝RAN切片资源分配问题。已有不少学者采用凸优化等传统优化方法^[10-13]对RAN切片资源分配问题展开研究。然而,在6G网络中,RAN切片所面临的业务资源需求差异化愈发显著,且网络环境受业务量的波动和信道状态的时变等因素影响,表现出高度动态变化特征。这使切片资源分配问题呈现出高维度与非线性的复杂特性,传统的数学模型方法难以精准描述下一代无线通信网络的复杂场景,故不再适用于6G RAN切片资源分配^[11]。深度强化学习 (DRL, deep reinforcement learning) 等人工智能方法通过与环境互动提取有效信息,可不断更新策略实现优化^[11,14]。因此,将RAN切片资源分配与人工智能相结合^[15],能根据无线通信网络场景的动态变化,自适应地调整资源分配策略,及时响应用户终端需求^[13-14]。

近年来,已有相关论文基于人工智能方法研究RAN切片资源分配问题。文献[16]探讨了深度Q学

习 (DQL, deep Q-learning) 在5G网络切片资源分配中的应用,通过将用户需求与切片资源精准匹配,提升了资源分配的有效性。文献[17]将离散归一化优势函数引入DQL中,分离Q函数为状态价值函数和优势函数,结合确定性策略梯度下降 (DPGD, deterministic policy gradient descent) 算法提升了收敛速度。然而上述2篇文献中的算法只局限于切片内部面向用户的资源分配,切片的资源量是固定的,并未考虑切片总体资源的配置,本质上属于完全硬隔离切片分配方案^[18]。文献[19]结合双层深度Q网络 (DDQN, double deep Q-network) 提出了5G网络中一种能感知业务需求的资源分配算法,以各切片的实时业务申请量为依据,实现对切片的动态资源配置。文献[20]提出了5G RAN切片之间共享资源的方法,从总资源池中提取部分资源给RAN切片进行资源调整,能够有效应对终端用户设备 (UE, user equipment) 服务需求的变化。以上2篇文献虽然不再固定切片的资源量,但是切片资源重配置是以用户需求变化而触发的,存在重配置次数过多的问题,增加了算法运算复杂度。

鉴于网络环境可能受业务流量长期动态变化以及信道状态瞬时变化的影响^[21],研究者提出了一种多时间尺度协同优化的网络切片资源分配算法。文献[22]以响应5G网络切片业务流变化为出发点,提出了一个深度学习 (DL, deep learning) 与DRL协作框架,将资源分配划分为大时间尺度和小时间尺度2个阶段,其中大时间尺度采用DL执行面向网络切片的资源配置,小时间尺度采用DRL执行面向用户的资源分配。文献[23]沿用文献[22]的框架,选择双向长短期记忆 (LSTM, long short-term memory) 网络在大时间尺度上预测资源需求量,并使用异步优势动作评价 (A3C, asynchronous advantage actor-critic) 算法在小时间尺度上进行用户接纳决策,然而没有考虑面向用户的资源分配,且并未考虑多类型业务的QoS约束。文献[24]同样采取多时间尺度协同的框架,以最大化网络长期效用和频谱利用率为目标,在大小时间尺度上分别采取DDQN和深度确定性策略梯度 (DDPG, deep deterministic policy gradient) 算法执行资源的配置,然而所有切片资源分配均由同一智能体决策,难以保障不同类型用户的QoS需求。尽管文献[22-24]提出的多时间尺度协同优化的网络切片资源分配算法在

一定程度上实现了切片资源的周期性重配置, 但该重配置仅在大时间尺度上进行, 而切片资源量在每个小时间尺度保持固定, 在面对用户服务需求快速变化的场景时可能出现资源利用率低下或资源供给不足的问题^[11]。文献[25]提出了一种以业务需求为导向的 2 层 5G RAN 切片资源交易框架, 通过 DDQN 支持各切片在小时间尺度进行资源交易, 从而实现切片资源在小时间尺度的动态调整, 但该方法仅以移动网络运营商收益最大化为目标决定优先满足哪类业务需求, 并没有考虑终端侧的 QoS 需求。

目前, 6G 无蜂窝 RAN 切片资源分配问题研究仍处于起步阶段。文献[26]以最小化时延为优化目标, 提出了基于非正交多址接入的 2 层 RAN 切片资源分配框架, 采用多智能体强化学习 (MARL, multi-agent reinforcement learning) 算法进行面向用户的资源分配, 并通过实时观测用户需求变化重配置切片资源。然而, 该算法仅考虑单一类型业务, 并且将 2 层资源分配在同一个时间尺度内优化, 导致切片重配置次数过多, 增加了算法运算复杂度。文献[27]研究了 6G 无蜂窝场景中用户与 AP 的连接方式、功率分配和资源分配的联合优化问题, 以最大化系统效用为目标, 运用多智能体深度确定性策略梯度 (MADDPG, multi-agent deep deterministic policy gradient) 算法进行优化, 有效提高了用户速率。然而, 由于小时间尺度切片资源量固定, 该算法无法及时响应业务流的变化, 特别是应对突发业务时存在较大的局限性。

6G 网络环境呈现出高度动态变化的特性, 不仅业务类型多样, 还会出现诸如智能驾驶场景中检测到车轮胎突然漏气或工业物联网场景中生产线突发故障等突发业务状况, 因此, 6G RAN 切片资源分配不仅需满足各类业务差异化的 QoS 需求, 还需具备对突发业务状况进行高效资源调度的能力^[22]。基于此, 本文提出了一种基于多时间尺度协同优化的无蜂窝 RAN 切片资源分配 (RAMCO, resource allocation based on multi-timescale collaborative optimization) 算法, 该算法将动态调整思想融入大小时间尺度的资源分配中, 同时考虑增强型移动宽带 (eMBB, enhanced mobile broadband) 与超可靠低时延通信 (uRLLC, ultra-reliable and low-latency communication) 两类业务的

QoS 需求。此外, 为了应对切片资源需求的不可预知变化, 本文通过共享预留资源辅助切片资源的分配与动态调整。尽管此前, 文献[20]和文献[22]均在切片资源短缺时向共享资源索取支持, 但文献[20]在配置各切片资源和共享资源时以最大化服务等级协议满足率和资源利用率为目标, 因此即使切片内用户的服务质量都已得到满足, 切片仍会占用共享资源, 从而降低了切片隔离度。此外, 文献[20]在为各切片配置资源时并没有考虑面向用户的资源分配。文献[22]中大时间尺度的资源配置没有考虑各类业务的时延等 QoS 约束, 从而导致资源需求预测不准确, 会出现某类切片占用过多共享资源和减弱切片隔离性能的问题。此外, 在小时间尺度缺乏共享资源分配的优化方案, 对于突发到达的业务流存在资源调整的滞后性。本文算法赋予共享资源智能决策能力, 使其能够根据用户实时状态信息做出分配资源的决策。同时, 随着在大时间尺度对切片资源需求的精准预测, 并实施周期性资源重配置, 各切片对共享资源的依赖程度会逐渐降低, 进而保障切片的隔离性能。本文的主要贡献总结如下。

1) 以最大化无蜂窝 RAN 系统效用为目标, 以 eMBB 和 uRLLC 两类业务传输速率和平均时延等 QoS 需求为约束, 提出了多时间尺度协同优化的切片资源分配算法。

2) 将大时间尺度面向切片的资源配置问题构建为以用户平均时延为约束的基于均方误差 (MSE, mean square error) 最小化的资源需求预测模型, 根据业务流量预测 eMBB 和 uRLLC 切片所需资源, 并通过长短期记忆时序预测网络实现切片资源的精准配置。

3) 将小时间尺度面向用户的资源分配问题构建为以最大化系统效用, 同时保障用户传输速率为目标的资源分配模型, 通过设定 eMBB、uRLLC 切片管理智能体^[27] (简称切片智能体) 和共享资源智能体, 提出了一种集中训练分布执行框架下基于多智能体近端策略优化 (MAPPO, multi-agent proximal policy optimization) 的资源分配算法, 协同完成资源的分配与调整。

4) 通过仿真实验评估本文算法的性能, 并与已有算法进行比较。结果表明本文算法能够实现切片资源需求的准确预测, 并且通过切片周期性的重

配置以及多智能体之间的协作资源分配,能够有效提高系统传输速率,降低用户平均时延和业务阻塞率。

1 系统模型

本文考虑的 6G 无蜂窝 RAN 架构^[28]如图 1 所示,该架构包含边缘分布式单元 (EDU, edge distributed unit)、以用户为中心的分布式单元 (UCDU, user-centric distributed unit) 和中央处理单元 (CPU, central processing unit)。其中, EDU 负责管理其区域内的 AP, 实现分布式预编码, UCDU 实现数据的分发与整合, CPU 作为控制核心, 负责管理设备的配置, 执行物理资源块 (PRB, physical resource block) 分配的决策。该架构通过大量部署 AP 消除传统蜂窝网络架构的边界效应, 已有研究表明全连接方式能实现最优服务质量^[2,29], 因此本文考虑用户与 AP 全连接的接入方式提升系统性能。假设网络场景中存在 N 个用户和 J 个 AP, 每个 AP 配备 J_L 根天线, 用户集合为 $\mathcal{N} = \{1, 2, \dots, n, \dots, N\}$, AP 集合为 $\mathcal{J} = \{1, 2, \dots, j, \dots, J\}$ 。所有切片集合表示为 $\mathcal{I} = \{1, 2, \dots, i, \dots, I\}$ 。根据不同业务将网络切片划分为 eMBB 切片 $\mathcal{I}^{\text{eMBB}}$ 和 uRLLC 切片 $\mathcal{I}^{\text{uRLLC}}$, 且满足 $\mathcal{I} = \mathcal{I}^{\text{eMBB}} \cup \mathcal{I}^{\text{uRLLC}}$, 切片 i 中的所有用户集合表示为 \mathcal{U}_i 。在实际部署中, 为避免切片中用户数量过多造成导频污染, 进而影响用户的信干噪比 (SINR, signal to interference plus noise ratio), 通常将同一切片内的用户划分为多个协作簇^[30], 用集合 $\mathcal{M}_i = \{1, 2, \dots, m, \dots, M_i\}$ 表

示切片 i 中的协作簇, 同一协作簇内的用户将共享相同的时频资源。已有论文使用图论、凸优化等方法研究协作簇的划分^[10,29], 由于本文重点关注无蜂窝 RAN 切片资源分配问题, 因此假设切片中用户已划分至各协作簇。

在无蜂窝 RAN 场景中, 通常采用正交频分复用 (OFDM, orthogonal frequency division multiplexing) 技术将总频谱资源划分为 F 个带宽相同的 PRB, 其中每个 PRB 的带宽为 $B = \frac{B^{\text{max}}}{F}$ 。总频谱资源分为切片资源和共享资源, 满足 $F = \sum_{i \in \mathcal{I}} \theta_i + \theta_s$, 其中 θ_i 和 θ_s 分别为切片资源量和共享资源量。该场景采用时分双工的工作模式, 每个相干间隔 τ_c 分为 3 个信号传输阶段, 分别为上行导频训练阶段、上行链路数据传输阶段和下行链路数据传输阶段^[2]。鉴于上下行数据传输的目的与要求通常不一致^[31], 一般只考虑 AP 为用户终端提供服务的下行链路数据传输阶段^[10,27]。在上行导频训练阶段, 用户将向所有 AP 发送至特定的导频序列。待各 AP 接收后, 再进行信道估计, 从而获得信道状态信息 (CSI, channel state information)。假设正交导频序列长度为 τ , 定义切片 i 中协作簇 m_i 中用户 n 向所有 AP 发送的导频序列为 $(\tau)^{\frac{1}{2}} \zeta_{i,m,n} \in \mathbb{C}^{\tau \times 1}$, 满足 $\|\zeta_{i,m,n}\|^2 = 1$ 。假设同一协作簇内不同用户之间的导频序列具有正交性, 可得到传输时间间隔 (TTI, transmission time interval) t 时 AP j 在估计用户 n 信道时接收到的有效信号为^[32]

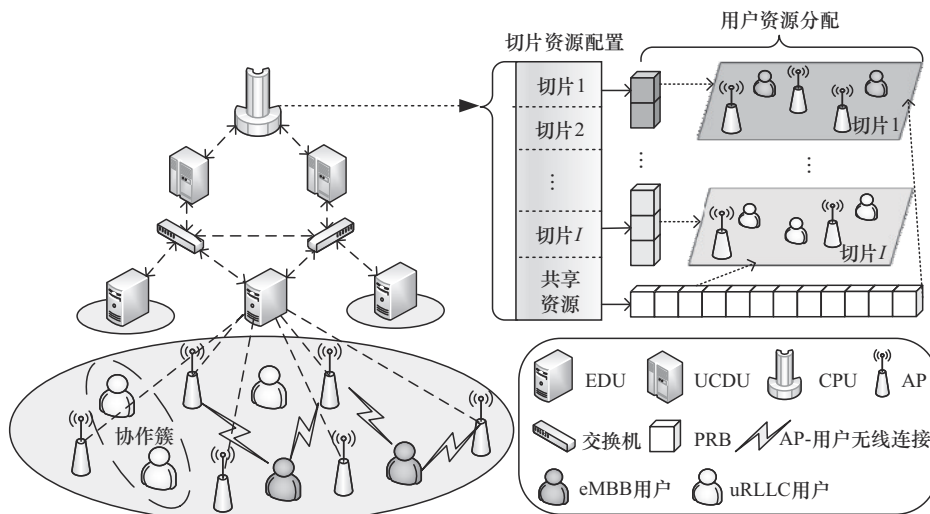


图 1 6G 无蜂窝 RAN 架构

$$y_{i,m,n,t}^j = \sqrt{\tau p^p} g_{i,m,n,t}^j + o_{i,m,n,t}^j \quad (1)$$

其中, p^p 表示每个导频序列归一化后的发射功率, $o_{i,m,n,t}^j \in \mathbb{C}^{J_L \times \tau}$ 表示服从复高斯分布的加性噪声, $g_{i,m,n,t}^j$ 表示协作簇 m_i 中用户 n 与 AP j 之间的信道增益, 可表示为

$$g_{i,m,n,t}^j = \sqrt{\beta_{i,m,n,t}^j} h_{i,m,n,t}^j \quad (2)$$

其中, $\beta_{i,m,n,t}^j$ 表示 AP 与用户之间的大尺度衰落, 涵盖路径损耗与阴影效应的影响, $h_{i,m,n,t}^j$ 表示小尺度衰落特性。对式(1)采用最小均方误差方法进行信道估计, 可估计出用户 n 与 AP j 之间的信道增益为

$$\hat{g}_{i,m,n,t}^j = \frac{\sqrt{\tau p^p} \beta_{i,m,n,t}^j}{1 + \tau p^p \beta_{i,m,n,t}^j} y_{i,m,n,t}^j \quad (3)$$

在下行链路数据传输阶段, AP 会根据估计的信道增益进行预编码处理^[33]获得预编码矢量 $v_{i,m,n,t}^j$, 并根据 CPU 设定的功率系数 $p_{i,m,n,t}^j$ 和用户 n 在 TTI t 的传输数据 $s_{i,m,n,t}$, 得到 AP j 向用户 n 发送的信号 $x_{i,m,n,t}^j$, 计算式为

$$v_{i,m,n,t}^j = \frac{\hat{g}_{i,m,n,t}^j}{\sqrt{\mathbb{E} \left[\left\| \hat{g}_{i,m,n,t}^j \right\|_2^2 \right]}} \quad (4)$$

$$R_{i,n,t} = \begin{cases} \frac{\tau_c - \tau}{\tau_c} \sum_{m \in \mathcal{M}_i} \alpha_{i,m,n} B_{i,m,n,t} \text{lb} \left(1 + \Gamma_{i,m,n,t} \right), & \text{eMBB 切片} \\ \frac{\tau_c - \tau}{\tau_c} \sum_{m \in \mathcal{M}_i} \alpha_{i,m,n} B_{i,m,n,t} \left[\text{lb} \left(1 + \Gamma_{i,m,n,t} \right) - \sqrt{\frac{V_{i,m,n,t}}{\rho_i}} \frac{Q^{-1}(\varpi)}{\ln 2} \right], & \text{uRLLC 切片} \end{cases} \quad (11)$$

根据香农定理和有限长块码理论^[34]可以分别求出 eMBB 切片与 uRLLC 切片中用户的传输速率, 可用式(11)表示。其中 $B_{i,m,n,t}$ 表示用户带宽, $\alpha_{i,m,n} \in \{0,1\}$ 表示切片 i 中的用户 n 是否属于协作簇 m ($m \in \mathcal{M}_i$), 并满足 $\sum_{m \in \mathcal{M}_i} \alpha_{i,m,n} = 1, \forall i \in \mathcal{I}, \forall n \in \mathcal{U}_i$, $Q^{-1}(\cdot)$ 表示逆高斯 Q 函数, ϖ 表示有效解码错误概率, ρ_i 表示 uRLLC 切片传输的平均数据包长度, $V_{i,m,n,t}$ 表示信道弥散, 具体表示为

$$V_{i,m,n,t} = 1 - (1 + \Gamma_{i,m,n,t})^{-2} \quad (12)$$

本文考虑的数据包时延包含传输时延和排队时延两部分。具体而言, 传输时延取决于式(11)中的用户传输速率, 排队时延取决于数据包到达率和传输速率。假设每个用户都有一个虚拟数据队列来缓冲传入的数据包, 并遵从先到先服务的策略传输。

$$x_{i,m,n,t}^j = \sqrt{p_{i,m,n,t}^j} v_{i,m,n,t}^j s_{i,m,n,t} \quad (5)$$

用户 n 从所有相连的 AP 接收到的信号可表示为

$$y_{i,m,n,t} = \sum_{j \in \mathcal{J}} g_{i,m,n,t}^{j,H} x_{i,m,n,t}^j + o_{i,m,n,t} = \text{DS}_{i,m,n,t} s_{i,m,n,t} + \text{NI}_{i,m,n,t} s_{i,m,n,t} + o_{i,m,n,t} + \sum_{n' \in m_i, n' \neq n} \text{UI}_{i,m,n',t} s_{i,m,n',t} \quad (6)$$

其中, $\text{DS}_{i,m,n,t}$ 表示期望信号, $\text{NI}_{i,m,n,t}$ 表示预编码增益的不确定性, $\text{UI}_{i,m,n',t}$ 表示同一协作簇中其他用户的干扰, 具体表示为

$$\text{DS}_{i,m,n,t} = \mathbb{E} \left[\sum_{j \in \mathcal{J}} \sqrt{p_{i,m,n,t}^j} g_{i,m,n,t}^{j,H} v_{i,m,n,t}^j \right] \quad (7)$$

$$\text{NI}_{i,m,n,t} = \sum_{j \in \mathcal{J}} \sqrt{p_{i,m,n,t}^j} g_{i,m,n,t}^{j,H} v_{i,m,n,t}^j - \text{DS}_{i,m,n,t} \quad (8)$$

$$\text{UI}_{i,m,n',t} = \sqrt{p_{i,m,n',t}^j} g_{i,m,n,t}^{j,H} v_{i,m,n',t}^j \quad (9)$$

因此, 用户的 SINR 可以表示为

$$\Gamma_{i,m,n,t} = \frac{|\text{DS}_{i,m,n,t}|^2}{\mathbb{E} \left[|\text{NI}_{i,m,n,t}|^2 \right] + \sum_{n' \in m_i, n' \neq n} \mathbb{E} \left[|\text{UI}_{i,m,n',t}|^2 \right] + 1} \quad (10)$$

使用变量 $A_{i,n,t}$ 表示切片 i 中用户 n 的数据包在 TTI t 的到达过程, 假设该到达过程在每个 TTI 独立同分布, 且服从泊松分布, 到达率 $\lambda_{i,n,t} = \mathbb{E} [A_{i,n,t}]$ 。使用变量 Ω_i 表示切片 i 中用户的平均包长, Δt 表示每个 TTI 的时长, $Q_{i,n,t}$ 表示切片 i 中用户 n 在 TTI t 的队列长度, 队列更新过程可表示为^[35-36]

$$Q_{i,n,t+1} = \max \left\{ Q_{i,n,t} - \frac{R_{i,n,t} \Delta t}{\Omega_i}, 0 \right\} + A_{i,n,t} \quad (13)$$

2 问题定义与建模

本文采用多时间尺度协同优化的资源分配算法框架。为了向不同时间尺度的资源分配策略提供明确的时间度量基准, 将大时间尺度的基本时间单位定义为 PW (prediction window), 小时间尺度的基本时间单位定义为 TTI, 假设每个 PW 包含 ΔT 个

TTI, 即 $PW = \Delta T$ (TTI)。如图2所示, 在大时间尺度的每个PW的开始阶段, 执行面向切片的资源配置操作。在该PW期间, 各切片所配置的资源量将固定不变, 以保障切片的隔离性能, 直至下一个PW才会对各切片资源进行重新配置。在小时间尺度的每个TTI, 各切片依据已配置的资源, 向用户进行资源分配, 以应对不同TTI的动态业务需求。而当切片向用户分配的资源不足以满足其服务质量需求时, 共享资源将参与到资源分配过程中。

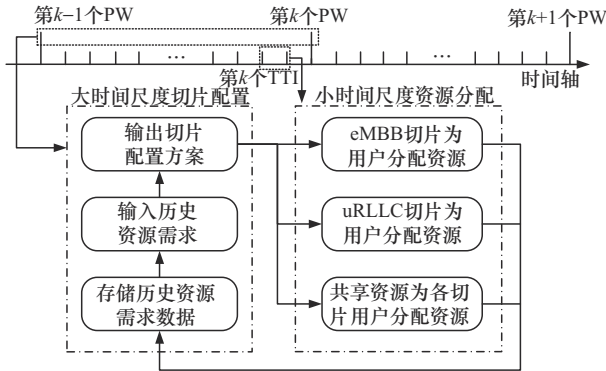


图2 多时间尺度协同优化算法框架

2.1 大时间尺度的优化问题

鉴于切片资源需求量呈现动态变化的特性, 本文利用资源的历史消耗数据, 对下一个PW各切片的资源需求进行预测。同时, 将大时间尺度用户的平均时延作为约束条件, 以保证切片配置方案的有效性。因此, 大时间尺度的切片资源配置问题可转换为一个PW内最小化切片资源需求预测值 $\tilde{A}_i(t)$ 与实际资源消耗值 $A_i(t)$ 之间的均方误差, 并满足平均时延约束的优化问题。均方误差越小, 说明预测精度越高, 资源需求的预测结果越符合实际资源需求, 越能降低切片对共享资源的依赖, 提升切片的隔离性能。该优化问题可表示为

$$\begin{aligned} & \arg \min_{\tilde{A}_i(t)} \frac{1}{\Delta T} \sum_{t=(k-1)\Delta T+1}^{k\Delta T} |A_i(t) - \tilde{A}_i(t)|^2 \\ \text{s.t. } C_1: & \sum_{i \in \mathcal{I}} \tilde{A}_i(t) = \Theta_k \leq F, t \in [(k-1)\Delta T + 1, k\Delta T], \\ & k = 1, 2, \dots \\ C_2: & \bar{D}_{i,n,k} \leq D_i^{\max}, k = 1, 2, \dots \end{aligned} \quad (14)$$

其中, C_1 约束了在PW k 期间各切片配置资源量总和 Θ_k 不可超过总资源量 F ; C_2 约束了PW k 期间各切片内用户的平均时延不可超过阈值 D_i^{\max} , 保障了用户在数据包时延方面的服务质量需求。

根据文献[36]中的速率稳定定理, 当所有离散队列 $Q_{i,n,t}$ 满足均值速率稳定性时, 即 $\lim_{t \rightarrow \infty} \frac{E\{Q_{i,n,t}\}}{t} = 0$, 网络处于稳定状态。则可以根据Little定理[37]推导出切片 i 中的用户 n 在PW k 的平均排队时延, 表示为

$$\bar{D}_{i,n,k}^q = \frac{\bar{Q}_{i,n,k}}{\bar{\lambda}_{i,n,k}} \quad (15)$$

其中, $\bar{Q}_{i,n,k} = \frac{1}{\Delta T} \sum_{t=(k-1)\Delta T+1}^{k\Delta T} E[Q_{i,n,t}]$ 是切片 i 中用户 n 在PW k 的平均队列长度, $\bar{\lambda}_{i,n,k} = \frac{1}{\Delta T} \sum_{t=(k-1)\Delta T+1}^{k\Delta T} \lambda_{i,n,t}$

是用户 n 在PW k 的数据包平均到达率。因此, 可以推导出平均时延计算式为

$$\begin{aligned} \bar{D}_{i,n,k} &= \bar{D}_{i,n,k}^{\text{trans}} + \bar{D}_{i,n,k}^q = \\ & \frac{1}{\Delta T} \sum_{t=(k-1)\Delta T+1}^{k\Delta T} \left(\frac{Q_i}{R_{i,n,t}} + \frac{E[Q_{i,n,t}]}{\lambda_{i,n,t}} \right) \end{aligned} \quad (16)$$

在每个PW中, 可能存在以下2种情况: 1) 实际资源消耗值未超过资源需求预测值, 即 $A_{i,k}(t) \leq \tilde{A}_{i,k}(t)$, 则在该PW期间, 该切片的资源量将维持不变从而保持切片之间的隔离性能; 2) 实际资源消耗值大于资源需求预测值, 即 $A_{i,k}(t) > \tilde{A}_{i,k}(t)$, 则共享资源会协助完成资源分配, 并在下一个PW开始时, 对该切片配置更多资源, 减少对共享资源的依赖。

2.2 小时间尺度的优化问题

在小时间尺度, 若因为业务需求的动态变化而频繁重配置切片资源, 不仅会给系统增加额外计算开销, 也会削弱切片的隔离性能。所以当各切片完成当前PW的资源配置后, 会根据已配置的资源在每个TTI执行面向用户的资源分配。本文考虑了用户的传输速率作为小时间尺度资源分配的约束, 并定义了切片的效用函数 $U_{i,k}$, 具体表示为

$$U_{i,k} = \kappa_{i,1} \frac{1}{\Delta T} \sum_{t=(k-1)\Delta T+1}^{k\Delta T} \sum_{n \in \mathcal{U}_i} R_{i,n,t} - \kappa_{i,2} \left(\sum_{n \in \mathcal{U}_i} \bar{D}_{i,n,k} \right) \quad (17)$$

其中, $\kappa_{i,1}$ 和 $\kappa_{i,2}$ 是权重因子。因此, 小时间尺度的资源分配优化目标函数可具体描述为

$$\begin{aligned} & \max \sum_{i \in \mathcal{I}} U_{i,k} \\ \text{s.t. } C_1: & \sum_{m \in \mathcal{M}_i} a_{i,m,t} \leq \tilde{A}_{i,k}, \forall i \in \mathcal{I}, t \in [(k-1)\Delta T + 1, \\ & k\Delta T], k = 1, 2, \dots \end{aligned}$$

$$\begin{aligned}
C_2: & \sum_{i \in \mathcal{I}} \sum_{m \in \mathcal{M}_i} b_{i,m,t} \leq \Theta_{s,k}, \forall t \in [(k-1)\Delta T + 1, k\Delta T], \\
& k = 1, 2, \dots \\
C_3: & a_{i,m,t} + b_{i,m,t} \geq 1, \forall i \in \mathcal{I}, \forall m \in \mathcal{M}_i \\
C_4: & R_{i,n,t} \geq R_i^{\min}, \forall i \in \mathcal{I}, n \in \mathcal{U}_i
\end{aligned} \quad (18)$$

其中, C_1 约束了在每个 TTI 各切片向用户所在协作簇分配的资源量 $a_{i,m,t}$ 总和不可超过该 PW 期间切片预配置的资源量 $\tilde{A}_{i,k}$; C_2 约束了用户分得的共享资源量 $b_{i,m,t}$ 总和不可超过该 PW 期间预配置的共享资源量 $\Theta_{s,k}$; C_3 约束了每个用户协作簇至少占用一块 PRB; C_4 约束了所有用户的传输速率不可低于其所在切片的最小速率 R_i^{\min} , 该阈值依据切片类型进行设置。

3 算法设计

从优化问题式(14)和式(18)可以看出, 本文提出的 6G 无蜂窝 RAN 场景中的切片资源分配问题属于 NP-hard 问题。此外, 无线信道状态和业务需求也会随着时间动态变化, 这使传统的静态优化算法难以契合该场景需求。因此, 本文引入人工智能方法, 旨在解决复杂环境下多时间维度的资源分配问题。

3.1 大时间尺度的算法设计

为解决大时间尺度的切片资源配置问题, 本文使用基于 LSTM 的预测方法。LSTM 是一种递归神经网络, 具有解决预测时间序列问题的能力。其通过引入遗忘门、输入门和输出门控制信息的写入、读取和重置操作, 能够有效地存储和更新长期依赖信息, 适应资源需求的动态变化^[23], 其架构如图 3 所示。在本文中, 存储历史 PW 各 TTI 的切片 i 内用户的资源实际消耗量 $A_i(t) = \sum_{m \in \mathcal{M}_i} (a_{i,m,t} + b_{i,m,t})$,

用 $\mathcal{A}_i = \{A_i(t-2PW), \dots, A_i(t-PW), \dots, A_i(t)\}$ 表示 LSTM 输入数据。值得注意的是, 在存储 $A_i(t)$ 前还需要进行数据预处理, 将通过式(14)中约束条件 C_2 判定该历史资源分配方案是否满足用户的平均时延指标, 若满足则直接存储, 若不满足, 则采用二分法从共享资源中寻找能够满足用户平均时延指标的 PRB 量, 并加上原资源分配量作为修正值进行存储。输出的下一个 PW 预测结果为 $\mathcal{P}_i = \{P_i(t+1), \dots, P_i(t+PW)\}$, 且预测结果的均值和方差分别如式(19)和式(20)所示。同时结合置信水平 $\chi \in (0, 1)$ 计算出下一个 PW 切片的配置值所处区间, 如式(21)所

示, 选取置信区间的上限作为下一个 PW 切片的资源配置值 $\tilde{A}_i(t)$ ^[22]。

$$\bar{P}_i = \frac{1}{PW} \sum_{t'=1}^{PW} P_i(t+t') \quad (19)$$

$$\sigma(P_i) = \sqrt{\frac{1}{PW} \sum_{t'=1}^{PW} (P_i(t+t') - \bar{P}_i)^2} \quad (20)$$

$$\tilde{A}_i(t) \in \bar{P}_i \pm z_{\frac{1-\chi}{2}} \frac{\sigma(P_i)}{\sqrt{PW}} \quad (21)$$

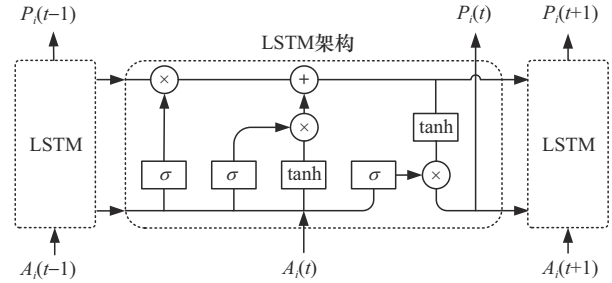


图3 大时间尺度的 LSTM 架构

尽管 LSTM 等深度学习方法能够实现长期趋势下稳定业务的精准预测, 然而在突发业务的预测上存在一定的局限性, 这是因为突发业务通常是随机的、出现频率低且波动性强的, 难以从有限的历史数据中捕捉突发业务发生的规律^[38]。随着系统运行时间增加, 突发业务出现频率增加, 当样本数据足够多时, LSTM 等深度学习方法能够逐渐学习到业务到达的潜在模式, 从而提高预测能力。为弥补 LSTM 在预测突发业务方面的不足, 本文在小时间尺度采用以切片资源优先和共享资源辅助的面向用户的资源分配策略, 以保障用户的 QoS 不受影响。

3.2 小时间尺度的算法设计

在小时间尺度, 本文聚焦于优化面向多种类型用户的资源分配策略, 以实现系统效用的最大化, 同时保障用户的服务质量需求。多智能体强化学习方法具备显著优势: 一方面, 它能通过与环境交互, 根据任务对象实时状态持续迭代优化策略; 另一方面, 其能够根据不同智能体的任务灵活设计奖励函数, 促使系统性能向预期目标发展, 尤其在处理非线性优化问题方面具有良好的适配度^[27]。因此, 本文引入了 MARL 框架, 并在 CPU 端部署切片智能体和共享资源智能体, eMBB 切片和 uRLLC 切片智能体执行各自管辖用户的资源分配, 共享资源智能体

则在必要情况下负责全体用户的资源调整。多智能体 $j=0,1,\dots,I$ 中共享资源智能体的标识为0, 切片智能体的标识为 $\{1,\dots,I\}$ 。一个伴随强化学习的决策过程通常被定义为马尔可夫决策过程(MDP, Markov decision process)。MDP通常以元组 $(S, \mathcal{O}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ 表示, 其中 S 表示全局状态空间, \mathcal{O} 表示智能体观察的局部状态空间, \mathcal{A} 表示动作空间, \mathcal{P} 表示状态转移函数, \mathcal{R} 表示奖励函数。在每个小时间尺度 t , 智能体根据状态 $s_t \in S$ 执行动作 $a_t \in \mathcal{A}$, 并根据状态转移函数 $\mathcal{P}(s_{t+1}|s_t, a_t)$ 转移到下一状态 s_{t+1} , 同时智能体收到环境反馈的奖励 r_t 。切片智能体与共享资源智能体的MDP基本元素的详细设置如下。

1) 状态

各智能体将关注各用户信道增益的变化和业务到达情况, 并根据观测结果进行无线资源的分配。因此, 切片智能体 i 在TTI t 的状态空间包括该切片内各用户的SINR与业务队列信息, 可表示为 $s_t^i = \{\Gamma_{i,t}, Q_{i,t}\}$ 。共享资源智能体的状态空间包含所有用户的SINR与业务队列信息, 可表示为 $s_t^0 = \{\Gamma_t, Q_t\}$ 。

$$\Gamma_{i,t} = \{\Gamma_{i,n,t} | n \in U_i\}, Q_{i,t} = \{Q_{i,n,t} | n \in U_i\} \quad (22)$$

$$\Gamma_t = \{\Gamma_{i,n,t} | \forall i \in \mathcal{I}, n \in U_i\}, Q_t = \{Q_{i,n,t} | \forall i \in \mathcal{I}, n \in U_i\} \quad (23)$$

2) 动作

切片智能体的动作是根据大时间尺度完成的切片配置资源向其内部用户的协作簇分配PRB, 即 $a^i(t) = [a_{i,1}(t), \dots, a_{i,m}(t), \dots, a_{i,M_i}(t)]$, 该动作空间需满足优化问题式(18)中的约束条件 C_1 , 即需满足分配的资源总量不超过切片当前的可用资源量 $\tilde{A}_{i,k}$ 。共享资源智能体的动作是 $a^0(t) = \{b_1(t), \dots, b_i(t), \dots, b_I(t)\}$, 其中 $b_i(t) = [b_{i,1}(t), \dots, b_{i,m}(t), \dots, b_{i,M_i}(t)]$, 该动作空间需满足优化问题式(18)中的约束条件 C_2 , 即需满足分配的资源总量不超过共享资源量 $\Theta_{s,k}$ 。值得注意的是, 各智能体协同后选择的动作也需要满足约束条件 C_3 , 从而保障每个用户所在的协作簇至少分配到1块PRB。当业务需求发生变化时, 如用户传输速率阈值 R_i^{\min} 发生改变, 智能体需要通过优化资源分配策略, 选择最合适的动作, 以满足优化问题式(18)中的约束条件 C_4 。

3) 奖励

由于eMBB切片和uRLLC切片智能体和共享

资源智能体的资源分配任务不同, 策略优化方向也不同, 因此各智能体的奖励函数不同, 需要分别设置。鉴于小时间尺度的优化目标为最大化系统效用, 其包含用户的传输速率与平均时延两部分, 平均时延包括传输时延和排队时延, 根据式(16)可知传输时延取决于用户的传输速率 $R_{i,n,t}$, 排队时延则取决于业务队列长度 $Q_{i,n,t}$, 因此奖励函数将围绕用户传输速率和业务队列长度设计。切片智能体的奖励函数 $r^i(t)$ 设置如式(24)所示。

$$r^i(t) = \begin{cases} \sum_{n \in U_i} R_{i,n,t}^a - \omega_{i,1} \lg(Q_{i,n,t} + 1), R_{i,n,t}^a \geq R_i^{\min} \\ -r^{\text{pen}}, R_{i,n,t}^a < R_i^{\min} \end{cases} \quad (24)$$

其中, $R_{i,n,t}^a$ 为用户从切片 i 分配的资源而实现的传输速率, 即将带宽 $B_{i,m,n,t} = a_{i,m,t}B$ 代入式(11)。当用户传输速率满足优化问题式(18)中约束条件 C_4 时给予正向奖励, 否则将受到惩罚, r^{pen} 为惩罚参数。正向奖励中增加了与业务队列相关的限制项, 其中 $\omega_{i,1}$ 为调节因子。这是因为业务队列长度会影响排队时延, 进而影响用户平均时延无法保障大时间尺度的时延约束 C_2 。因此, 为保障平均时延指标, 智能体需要依据队列排队长度调整资源分配决策。

当所有切片智能体向用户分配的资源不足以满足其传输速率和时延约束时, 需要共享资源智能体参与资源的分配, 其奖励函数 $r^0(t)$ 设置如式(25)所示, 其中 $R_{i,n,t}^b$ 为用户从共享资源智能体分配的资源而实现的传输速率, 即将带宽 $B_{i,m,n,t} = b_{i,m,t}B$ 代入式(11), 并且满足 $R_{i,n,t} = R_{i,n,t}^a + R_{i,n,t}^b$ 。可知存在以下4种情况: 1) 当切片智能体向用户分配的资源可满足其速率约束, 且共享资源智能体没有提供额外资源, 则正向奖励为0; 2) 当切片智能体向用户分配的资源满足其速率约束, 但共享资源智能体提供了该用户额外资源, 则会受到一个过度分配惩罚项 r^{over} ; 3) 当切片智能体向用户分配的资源无法满足速率约束, 且通过共享资源智能体分配资源保障了其速率指标, 此时会得到由速率指标决定的正向奖励, 以确保奖励值的上限, 同时引入由成本因子 δ 和分配的共享资源量决定的惩罚项, 从而抑制用户对共享资源使用的不合理; 4) 当切片智能体向用户分配的资源无法满足速率约束, 且通过共享资源智能体提供额外资源, 仍不能满足其传输速率指标, 这意味着共享资源分配量不足, 因此引入 R_i^{\min}

和 $R_{i,n,t}$ 之间的差值作为惩罚。需要注意的是，共享资源智能体的奖励函数设置也考虑了业务队列引起的排队时延，因此引入了与业务队列 $Q_{i,n,t}$ 相关的限制项， $\omega_{i,2}$ 为调节因子。可以看出，共享资源智能体奖励函数的设置，保障了当切片预配置资源不足时，通过向用户分配共享资源以满足用户服务质量需求，这与小时间尺度的优化目标是一致的。

本文采用 MAPPO 算法求解小时间尺度优化问题，并采用集中式训练分布式执行的框架，如图 4 所示。每个智能体 $j=0,1,\dots,I$ 都拥有一个动作网络

$$r^0(t) = \begin{cases} \sum_{n \in \mathcal{U}} -\omega_{i,2} \lg(Q_{i,n,t} + 1), R_{i,n,t}^a \geq R_i^{\min}, R_{i,n,t}^b = 0 \\ \sum_{n \in \mathcal{U}} -r^{\text{over}} - \omega_{i,2} \lg(Q_{i,n,t} + 1), R_{i,n,t}^a \geq R_i^{\min}, R_{i,n,t}^b > 0 \\ \sum_{n \in \mathcal{U}} R_i^{\min} - \delta \sum_{m \in \mathcal{M}_i} b_{i,m,t} - \omega_{i,2} \lg(Q_{i,n,t} + 1), R_{i,n,t}^a \leq R_i^{\min}, R_{i,n,t}^a + R_{i,n,t}^b \geq R_i^{\min} \\ \sum_{n \in \mathcal{U}} -(R_i^{\min} - R_{i,n,t}) - \omega_{i,2} \lg(Q_{i,n,t} + 1), R_{i,n,t}^a \leq R_i^{\min}, R_{i,n,t}^a + R_{i,n,t}^b < R_i^{\min} \end{cases} \quad (25)$$

$\text{loss}_i(\theta^i) =$

$$E_t \left[\min \left(\eta_t(\theta^i) \hat{A}_t^i, \text{clip} \left(\eta_t(\theta^i), 1 - \varepsilon, 1 + \varepsilon \right) \hat{A}_t^i \right) \right] \quad (26)$$

其中， $\eta_t(\theta^i)$ 是当前策略 $\pi_{\theta^i}^j$ 与旧策略 $\pi_{\theta_{\text{old}}^j}^j$ 之间的比值，如式(27)所示。

$$\eta_t(\theta^i) = \frac{\pi_{\theta^i}^j(a_t^i | s_t^i)}{\pi_{\theta_{\text{old}}^j}^j(a_t^i | s_t^i)} \quad (27)$$

优势函数可表示为

π^j 和一个评价网络 V^j ，动作网络作为策略网络指导智能体做决策，评价网络作为值网络评价动作网络质量。各智能体将本地状态信息 s_t^j 作为动作网络的输入，输出动作空间的概率分布，采样并执行动作 a_t^j ，获得奖励 r_t^j 反馈，紧接着将状态转移序列 $(s_t^j, a_t^j, r_t^j, s_{t+1}^j)$ 存储至经验回放池 \mathcal{D} 中，供评价网络 V^j 更新，并根据全局信息计算优势函数 \hat{A}_t^j ，指导动作网络 π^j 更新。该算法在策略梯度的基础上使用裁剪后的损失函数更新策略网络参数 θ^j ，从而避免更新幅度过大造成不收敛的问题，参数 θ^j 的损失函数如式(26)所示。

$$\hat{A}_t^j = \delta_t^j + \gamma v \delta_{t+1}^j + \dots + (\gamma v)^{T-t+1} \delta_T^j \quad (28)$$

其中， δ_t^j 是时序差分误差， γ 是折扣率， v 是平滑系数， $\text{clip}()$ 是裁剪函数，用于限制更新幅度，将 $\eta_t(\theta^i)$ 限制在 $[1 - \varepsilon, 1 + \varepsilon]$ 中，其具体表达式为

$$\delta_t^j = r_t^j + \gamma V(s_{t+1}^j) - V(s_t^j) \quad (29)$$

$$\text{clip}(\eta_t(\theta^i), 1 - \varepsilon, 1 + \varepsilon) = \begin{cases} 1 + \varepsilon, \eta_t(\theta^i) \geq 1 + \varepsilon \\ 1 - \varepsilon, \eta_t(\theta^i) < 1 - \varepsilon \end{cases} \quad (30)$$

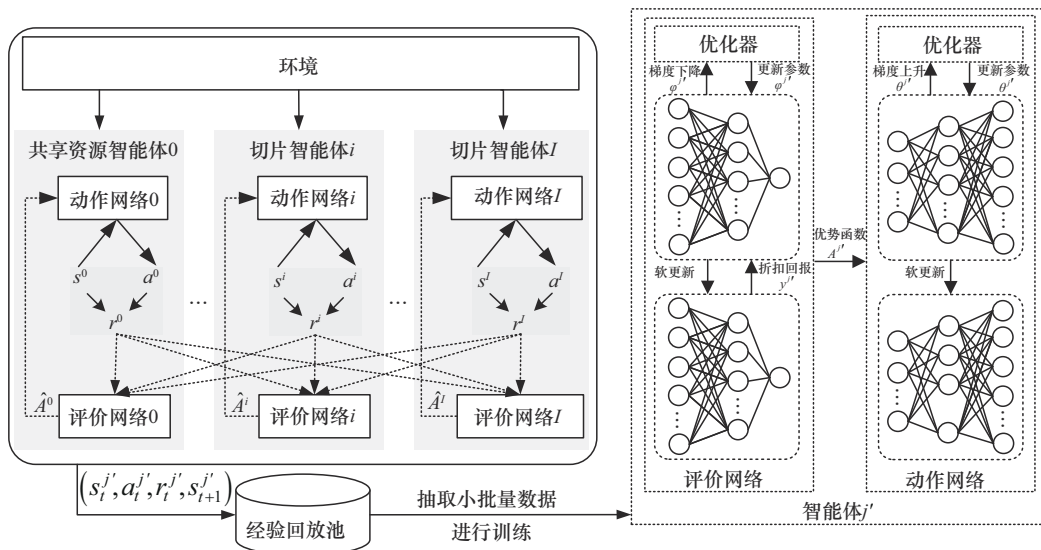


图 4 小时间尺度的 MAPPO 框架

评价网络则通过最小化损失函数来更新评价网络参数 φ^j , 其计算式为

$$\text{loss}_t(\varphi^j) = \mathbb{E} \left[\left(V_{\varphi^j}(s_t^j) - y_t^j \right)^2 \right] \quad (31)$$

其中, $V_{\varphi^j}(s_t^j)$ 是在状态 s_t^j 时评价网络输出的状态价值, y_t^j 是累计折扣回报值。

算法1具体描述了利用MAPPO算法进行小时间尺度资源分配的执行过程。步骤1)~步骤3)初始化系统参数, 设置了训练回合数 K_L 并初始化环境状态; 步骤4)~步骤11)是数据收集阶段, 各智能体根据初始动作网络 π_{θ}^j 与环境交互, 获得训练步长为 T_L 的轨迹 \mathcal{T}^j , 同时计算优势函数并将这些轨迹存储至经验回放池 \mathcal{D} 中; 步骤12)~步骤21)为训练阶段, 从经验回放池 \mathcal{D} 中抽取小批量数据, 并将轨迹分块, 在 Z 个训练轮次内迭代参数; 步骤22)~步骤23)为参数更新阶段, 用迭代后的新参数替换旧参数。在分布执行阶段, 各动作网络已训练完成, 不再部署评价网络, 每个智能体只需根据本地信息做出决策, 从而减少了时间消耗和计算资源消耗。

算法1 基于MAPPO的小时间尺度资源分配

1) 初始化: 各智能体动作网络 π_{θ}^j , 动作网络参数 θ_{old}^j , 评价网络 V_{φ}^j , 评价网络参数 φ_{old}^j , 经验回放池 \mathcal{D}

2) for episode=1,2,···, K_L

3) 所有智能体观察初始环境状态 s^j

4) for $t=1,2,···, T_L$

5) 各智能体根据策略 π_{θ}^j 分配 PRB

6) 根据式(11)计算各用户的传输速率, 并根据奖励函数计算奖励 r_t^j

7) 环境状态由 $s_t^j \rightarrow s_{t+1}^j$

8) end for

9) 获取该回合各智能体与环境的交互轨迹

$$\mathcal{T}^j = \left\{ s_t^j, a_t^j, r_t^j, s_{t+1}^j, \dots, s_{T_L}^j, a_{T_L}^j, r_{T_L}^j \right\}$$

10) 根据式(28)计算优势函数 \hat{A}_t^j

11) 将 $\left\{ [s_t^j, a_t^j, r_t^j, s_{t+1}^j, \hat{A}_t^j]_{j'=0}^{T_L} \right\}$ 存储至经验回放池 \mathcal{D} 中

12) for $z=1,2,···, Z$

13) 从经验回放池 \mathcal{D} 中抽取小批量数据, 并划分为长度为 E 的块

14) for $l'=0,1,···, T_L/E - 1$ do

$$15) \quad \mathcal{D}_{l'} = \left\{ [s_t^j, a_t^j, r_t^j, s_{t+1}^j, \hat{A}_t^j]_{j'=0}^{T_L} \right\}_{t=1+E'l'}$$

16) for 所有智能体 do

17) 根据式(26)计算损失函数 $\text{loss}(\theta^j)$, 对动作网络参数 θ^j 进行梯度上升更新

18) 根据式(31)计算损失函数 $\text{loss}(\varphi^j)$, 对评价网络参数 φ^j 进行梯度下降更新

19) end for

20) end for

21) end for

22) 各智能体更新旧动作网络参数 $\theta_{\text{old}}^j \leftarrow \theta^j$

23) 各智能体更新旧评价网络参数 $\varphi_{\text{old}}^j \leftarrow \varphi^j$

24) end for

3.3 多时间尺度协同的算法执行过程

本文提出的基于多时间尺度协同优化的无蜂窝RAN切片资源分配算法由大时间尺度的切片资源配置和小时间尺度的用户资源分配两部分组成。首先, 当系统初始化时, 设置切片数量、PRB总量及历史切片资源消耗数据, 以提供初始环境信息。在大时间尺度每个PW的开始阶段, 利用LSTM网络对各切片未来的资源需求进行预测, 从而给出初步的资源配置方案。随后, 在小时间尺度的每个TTI, 基于MAPPO算法训练多智能体, 使其根据用户信道增益和业务动态变化调整资源分配。当前PW结束后, 系统收集该时间段内每个TTI的实际资源使用数据, 将其作为新的输入反馈至LSTM模型, 以优化下一个PW的资源配置方案。

3.4 时间复杂度分析

LSTM 在每个PW的时间复杂度^[39]如式(32)所示。

$$\begin{aligned} O\left((4n_c n_c + 4n_i n_c + n_c n_o + 3n_c) T_e \Delta T\right) = \\ O(W T_e \Delta T) \end{aligned} \quad (32)$$

其中, n_c 是记忆单元数量, n_i 是输入单元数量, n_o 是输出单元数量, T_e 是训练轮次。

MAPPO 算法训练神经网络的时间复杂度^[40]如式(33)所示。

$$O\left(K_L T_L (I+1) \left(\sum_{l=0}^{L_{\text{actor}}} n_l^{\text{actor}} n_{l+1}^{\text{actor}} + \sum_{l=0}^{L_{\text{critic}}} n_l^{\text{critic}} n_{l+1}^{\text{critic}} \right)\right) \quad (33)$$

其中, n_l^{actor} 是动作网络在第 l 层的神经元数量, n_l^{critic} 是评价网络在第 l 层的神经元数量, L_{actor} 和 L_{critic} 分别是动作网络和评价网络的隐藏层数量。

由于通常 $K_L T_L > \Delta T$, 本文提出的 RA-MCO 算法的时间复杂度如式(34)所示。可以看出, 该算法时间复杂度只与内部神经网络架构有关, 增加用户数并不会影响算法复杂度。

$$O\left(K_L T_L \left(W T_e + (I+1) \left(\sum_{l=0}^{L_{\text{actor}}} n_l^{\text{actor}} n_{l+1}^{\text{actor}} + \sum_{l=0}^{L_{\text{critic}}} n_l^{\text{critic}} n_{l+1}^{\text{critic}} \right) \right) \right) \quad (34)$$

本文提出的 RA-MCO 算法在大时间尺度的时间复杂度与文献[22]提出的智能资源规划 (iRSS, intelligent resource scheduling strategy) 算法中大时间尺度的时间复杂度一致, 而由于在小时间尺度采用并行分布式架构, 从而显著降低算法运行时间。文献[27]结合 DDQN 与 MADDPG 算法的时间复杂度如式(35)所示。

$$O\left(K_L T_L (I+1) \left(\sum_{l=0}^{L_{\text{actor}}} n_l^{\text{actor}} n_{l+1}^{\text{actor}} + \sum_{l=0}^{L_{\text{critic}}} n_l^{\text{critic}} n_{l+1}^{\text{critic}} \right) + T_u \left(\sum_{l=0}^{L_{\text{DDQN}}} n_l^{\text{DDQN}} n_{l+1}^{\text{DDQN}} \right) \right) \quad (35)$$

其中, T_u 是上层训练步长。与本文算法相比, 仅在大时间尺度存在差异, 而在小时间尺度, MADDPG 与 MAPPO 算法的时间复杂度相同。

4 仿真分析

4.1 系统参数设置

本文在 Python3.9 和 PyTorch 环境中对提出的 RA-MCO 算法进行了仿真验证。仿真场景为覆盖范围 500 m×500 m 的无蜂窝多输入多输出 (MIMO, multiple input multiple output) 场景, 覆盖区域内的两类用户分别具有 eMBB 和 uRLLC 业务。该场景中的 RAN 切片类型有 eMBB 和 uRLLC 两类, 设定每类切片数量为 3 个, 并设置了共享资源池。具有 eMBB 业务需求的用户被划分至 eMBB 切片, 具有 uRLLC 业务需求的用户被划分至 uRLLC 切片。假设每个切片的活跃用户数相同, 在分配资源之前, AP 已连接了覆盖区域内的活跃用户, 并将相邻的同类型用户划分为协作簇。对于 MAPPO 算法中各动作网络和评价网络, 均为 3 层全连接层, 神经元个数均设置为 256、128 和 128,

每个训练回合步长为 $T_L = 100$ 。使用 Adam 作为优化器, 其学习率为 10^{-3} , 奖励的折扣率为 $\gamma = 0.9$ 。相关仿真参数设置参考文献[24,26-27], 具体仿真参数如表 1 所示。

表 1 仿真参数

参数	值
PW 持续时间/s	3 600
传输时间间隔 TTI/ms	1
LSTM 模型的置信水平 χ	0.97
AP 数量 J /个	5
每个 AP 的天线数量/根	64
每个切片的用户数/个	4, 4
每个切片的协作簇数/个	2, 2
PRB 数量/块	30
每块 PRB 的带宽/kHz	180
噪声功率谱密度/(dBm·Hz ⁻¹)	-174
AP 发射功率/dBm	20
eMBB 和 uRLLC 平均数据包长 Ω /B	800, 200
eMBB 和 uRLLC 数据包到达率/(packet·ms ⁻¹)	0.1, 0.02
eMBB 和 uRLLC 平均时延上限 D_i^{max} /ms	4, 2
eMBB 和 uRLLC 传输速率阈值 R_i^{min} /(kbit·s ⁻¹)	500, 200

4.2 对比算法介绍

本文选取静态资源分配 (SRA, static resource allocation) [41]、iRSS[22]及 MADDPG[27]算法作为对比算法, 以下是各对比算法的简要介绍。

1) SRA。该算法不考虑大小时间尺度协同与切片资源配置, 仅仅为每个用户分配相等的资源, 是一种不涉及人工智能的静态分配算法。

2) iRSS。该算法在大时间尺度采用 LSTM 进行切片资源配置, 在小时间尺度采用 A3C 算法对切片中用户进行资源分配。该算法没有考虑多类用户的服务质量需求, 虽涉及共享资源, 但没有考虑共享资源的分配决策。

3) MADDPG。该算法在大时间尺度采用 DDQN 进行切片资源配置, 在小时间尺度采用 MADDPG 算法对切片中用户进行 AP 连接、功率分

配和资源分配的优化。

4.3 仿真结果分析

图 5 展示了本文提出的 RA-MCO 算法和 iRSS 算法在大时间尺度 LSTM 预测的均方误差性能。通常情况下, PW 的长度要远大于训练回合, 因此 LSTM 可以在 PW 结束时收敛, 并提供最优的预测结果^[22], 本文设置 PW 的长度为 3 600 s。在该仿真中, 分别选取 70%、15% 和 15% 的数据用于训练、检验与测试。从图 5(a)可以看出, RA-MCO 算法的 MSE 大致在第 5 个训练回合开始收敛, 从局部放大图可看出其均方误差值在第 32 个训练回合靠近 0.000 7, 并逐渐趋于稳定。从图 5(b)可以看出, iRSS 算法的预测均方误差值收敛至

0.000 8 附近, 高于 RA-MCO 算法, 这是因为 iRSS 算法在预测资源需求时并未考虑时延等服务质量约束, 使得预测结果低于资源实际消耗值。

图 6 展示了本文提出的 RA-MCO 算法和 iRSS 算法在大时间尺度 LSTM 预测的误差频数性能。如图 6(a)所示, 资源需求预测值与实际资源消耗值间的误差集中在 -0.005 与 0.005 之间。如图 6(b)所示, iRSS 算法的资源需求预测值与实际资源消耗值间的误差集中在 0.005 与 0.015 之间, 这表明 RA-MCO 算法的 LSTM 预测精度更高, 可以用于预测资源需求, 实现周期性切片资源重配置。

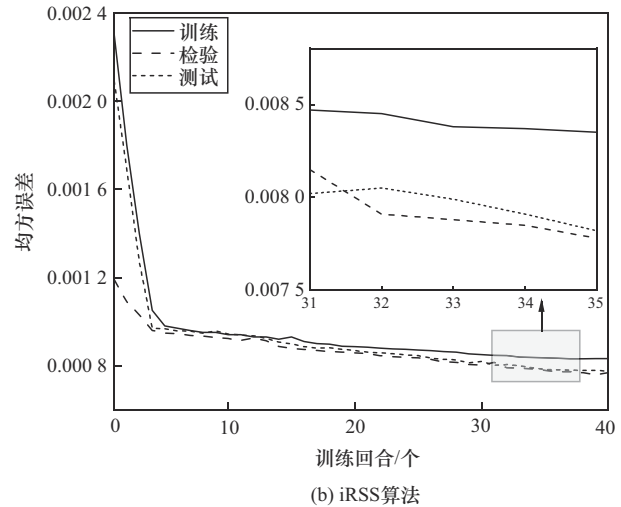
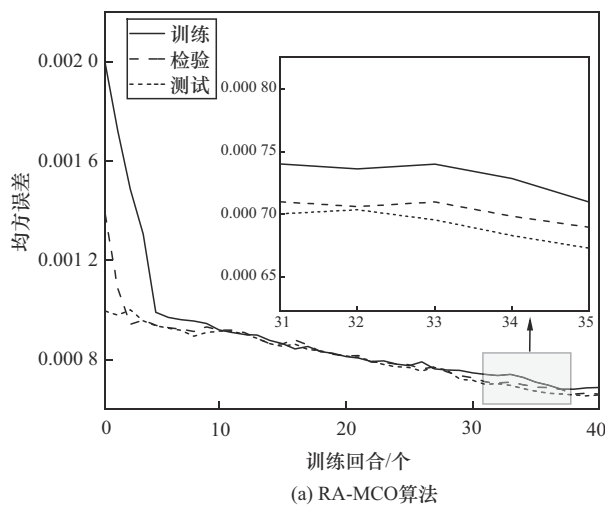


图 5 LSTM 预测均方误差性能

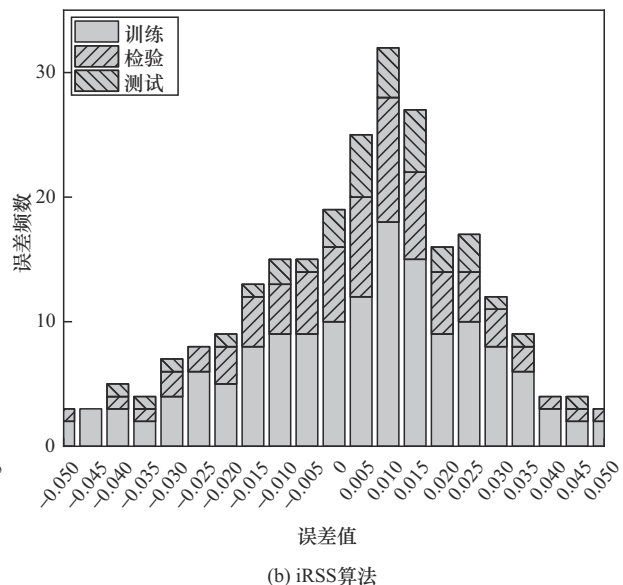
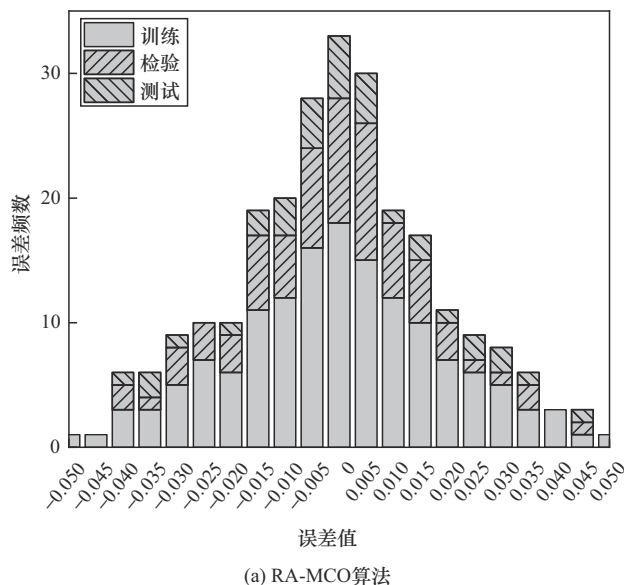


图 6 LSTM 预测误差频数性能

在小时间尺度 MAPPO 算法架构下, eMBB 和 uRLLC 这两类切片智能体以及共享资源智能体在不同学习率的训练过程中所获得的奖励情况如图 7 所示。由图 7(a)可知, 当两类切片配置的资源量和用户数分别相等时, eMBB 切片在训练期间所取得的奖励值高于 uRLLC 切片。原因在于 eMBB 类型用户的传输速率相对较高, 而 uRLLC 类型用户受信道弥散等因素影响, 其速率低于 eMBB 类型用户, 而奖励函数大小主要取决于用户传输速率。当学习率为 0.001 时, 在约 3 000 回合, 切片智能体的奖励值收敛至平稳状态。与之相比, 图 7(b)中的共享资源智能体在约 5 000 回合时才实现收敛。这是由于共享智能体的状态空间涵盖全体用户的状态信息, 相较于各切片智能体, 其动作空间更大, 故而收敛速度较慢。此外, 共享资源智能体的奖励函数涉及多个惩罚约束, 这也导致其奖励值相对较低。由图 7 可知, 当学习率减小至 0.000 1 时, 切片智能体和共享资源智能体分别在 4 000 回合与 6 000 回合左右收敛。这是因为随着学习率的降低, 策略更新速度略有下降, 但 2 种学习率最终均能收敛至近似相同的策略, 且收敛后的奖励基本一致。这表明本文算法对学习率参数具有一定的鲁棒性, 不易陷入局部最优或发散, 同时也说明所构建的多智能体学习框架具有良好的收敛稳定性。

切片的隔离性能是衡量其资源独立性和服务稳定性的重要指标, 通常表明在一定周期内, 其资源分配的稳定性及其保障内部用户性能不受资源变化影响的能力^[22-23]。本文用 $\kappa_{i,k}$ 表示切片 i 在 PW k 的

切片隔离性能, 可表示为

$$\kappa_{i,k} = \frac{\tilde{A}_{i,k}}{\tilde{A}_{i,k} + \sum_{t=1}^{\Delta T} \sum_{m \in \mathcal{M}_i} b_{i,m,t}} \quad (36)$$

其中, $\sum_{m \in \mathcal{M}_i} b_{i,m,t}$ 为共享资源在 PW k 向用户分配的 resource 量, $\kappa_{i,k}$ 越接近 1, 说明切片在该周期内依赖共享资源程度越低, 隔离性能越好。

图 8 展示了本文提出的 RA-MCO 算法和已有的 iRSS 算法的切片隔离性能。在 RA-MCO 算法中, 为实现系统效用最大化并保障业务的 QoS 要求, 切片尽可能地为用户分配更多的 PRB 资源, 直到该切片预配置的所有资源被分配完毕。若切片资源耗尽且用户 QoS 需求仍未满足时, 共享资源将提供额外支持。从图 8 可以看出, 在 RA-MCO 算法早期的 2 个 PW, 由于历史资源需求数据有限, 特别是突发业务的随机到达, 使大时间尺度的切片资源需求预测存在误差, 切片所获得的初始资源配置可能难以完全满足业务需求, 需要频繁地调用共享资源以满足服务质量要求, 此阶段切片隔离度较低。然而, 随着历史资源需求数据积累充分, 资源配置趋于精确, 切片对共享资源的依赖减少, 隔离性能逐步提升, 最终稳定在 100%。这表明此时预测结果与实际需求高度匹配, 各切片已能依靠配置资源独立完成服务质量保障, 从而降低了因频繁资源动态调整带来的系统开销。相比之下, iRSS 算法在资源预测方面并未考虑时延约束, 预测结果不够准确, 导致切片经常依赖共享资源。此外, 与 iRSS 算法相比,

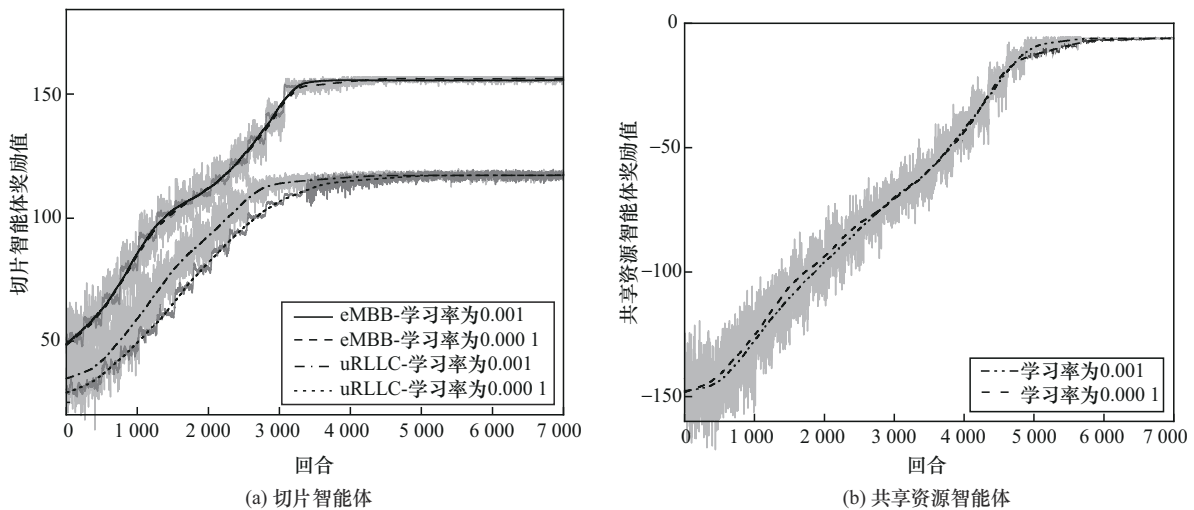


图 7 智能体在训练过程中所获奖励

RA-MCO算法中共享资源具备智能分配决策能力,能够基于用户的QoS需求和已分配的切片资源,实现向用户精准分配资源,从而提升了资源分配的稳定性,进而提升了切片隔离性能。因此,RA-MCO算法切片隔离性能始终高于iRSS算法。总之,本文算法通过结合大时间尺度的周期性切片重配置与小时间尺度的动态资源分配,不仅能够保障用户的QoS,还保障了切片的隔离性能。

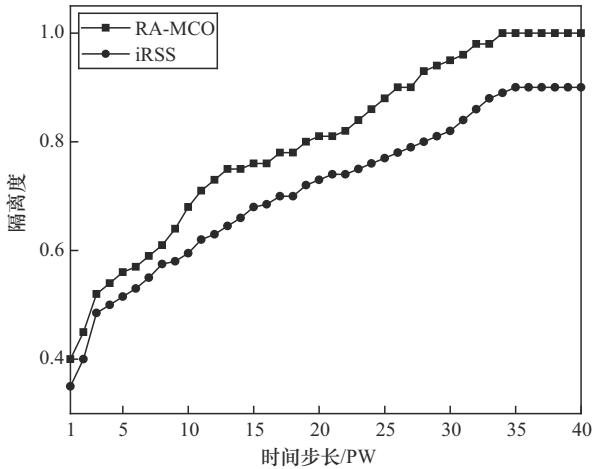


图8 切片隔离度随周期重配置的变化

以下评估RA-MCO算法对用户QoS的保障能力。图9展示了eMBB和uRLLC切片在不同算法中突发业务到达时的用户平均时延。由于突发业务通常是一个短暂的动态过程,本文设置在特定时间段内业务到达率突然增加,以模拟突发业务的到达^[38,42]。在图9中,在15~30 TTI期间,各切片的业务到达率上升至正常业务到达率的3倍,其余时间则维持正常水平。当突发业务到达时,排队数据包迅速堆积,用户队列长度显著增加,尤其是uRLLC切片的队列更容易在短时间内出现积压。在此情况下,RA-MCO算法能够通过启动共享资源为突发业务用户分配额外的PRB,从而提升传输速率,降低排队时延和传输时延,有效控制队列的迅速增长,保持队列稳定性,展现出了良好的鲁棒性。相比之下,SRA算法属于静态配置方案,所有用户可占用资源已固定,因此无法应对突发业务的到达,导致平均时延大幅上涨。iRSS算法中共享资源缺乏智能决策能力,难以实现精细化资源分配,因而在应对突发业务时效果有限。MADDPG算法采用固定的切片资源分配策略,通过优化用户与AP之间的连接关系提升

SINR,从而一定程度提高了用户传输速率,降低了时延增长幅度。然而,该算法在突发业务情况下的传输速率提升能力仍然受到切片资源总量的限制,难以完全适应突发业务流量的剧烈变化,平均时延仍然增加。当突发业务结束回到正常业务阶段时,各算法能及时处理排队数据,其平均时延恢复至先前正常业务时的水平。从图9可以看出,RA-MCO算法中eMBB切片的平均时延比SRA、iRSS和MADDPG算法分别平均降低了18.10%、9.46%和7.97%,uRLLC切片的平均时延比SRA、iRSS和MADDPG算法分别平均降低了35.54%、27.4%和24.47%。因此,RA-MCO算法在面对突发业务时表现出更优的适应性和鲁棒性,能够有效缓解业务流量突增带来的服务质量不能得到有效保障的问题。

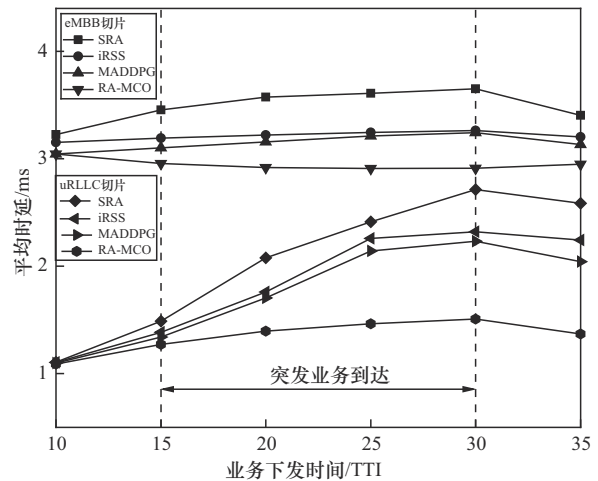


图9 突发业务到达时各算法平均时延对比

图10展示了不同算法在突发业务到达时系统的阻塞率性能,设置eMBB和uRLLC用户数均为20。在15~30 TTI期间,将各切片业务到达率提升至正常业务到达率的3倍以模拟突发业务的到达。在突发业务到达阶段,系统面临资源需求急剧上升和排队数据包快速积压的双重压力,容易造成资源分配不足而导致的业务阻塞。RA-MCO算法通过启动共享资源进行智能动态资源分配,保障各类型用户的传输速率和时延要求,其业务阻塞率最低,体现出良好的鲁棒性与自适应能力。相比之下,SRA算法由于无法动态调整资源,难以适应多用户的动态业务情况,其阻塞率最高。iRSS算法的共享资源虽能部分缓解突发业务,但缺乏环境感知能

力和智能决策能力,资源调整存在滞后性,无法精准分配资源,在突发业务阶段的阻塞率为20%~30%。MADDPG算法虽优化了AP与用户之间的连接关系与功率分配,提升了用户传输速率,但随着突发业务持续增加,仍存在资源紧张的情况,其阻塞率高于RA-MCO算法。从图10可以看出,RA-MCO算法的阻塞率比SRA、iRSS和MADDPG算法分别平均低75.31%、55.15%和39.14%。因此,本文提出的RA-MCO算法在业务动态变化时性能更为优越,其共享资源智能体能够根据业务量变化自适应调整资源分配,在确保用户QoS的同时降低业务阻塞率,增强了系统对复杂业务环境的适应能力。

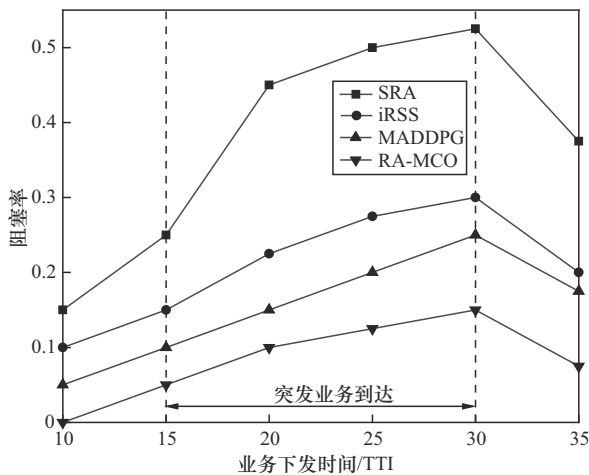


图10 突发业务到达时各算法阻塞率对比

图11显示了不同算法在改变用户传输速率阈值时的eMBB切片平均和速率性能。当传输速率阈值为0.4 Mbit/s时,各算法的eMBB切片平均和速率非常接近,这是因为此时仅靠切片内的资源即可满足用户传输速率要求。对于RA-MCO算法,当提升传输速率阈值时,切片资源不足以满足该服务质量要求,此时共享资源参与资源调整,提供额外的PRB帮助用户满足最低传输速率要求,因此eMBB切片平均和速率不断上升。当传输速率阈值为1 Mbit/s时,RA-MCO算法的eMBB切片平均和速率相比SRA、iRSS和MADDPG算法分别提升了64.7%、37.9%和24.3%。从图11可以看出,SRA算法不受传输速率阈值影响,因为其本质属于静态资源分配方案。iRSS算法虽有缓慢上升趋势,但受限于没有优化共享资源分配,因此无法满足所有用户的最低传输速率要求。MADDPG算法相较于

SRA与iRSS算法虽能通过更改AP与用户之间的连接关系以优化用户的信干噪比,但鉴于该优化存在上限,因此切片和速率仍受限于已配置的切片资源量。综上所述,RA-MCO算法通过共享资源智能体的动态资源调整,能够有效应对业务需求的动态变化。

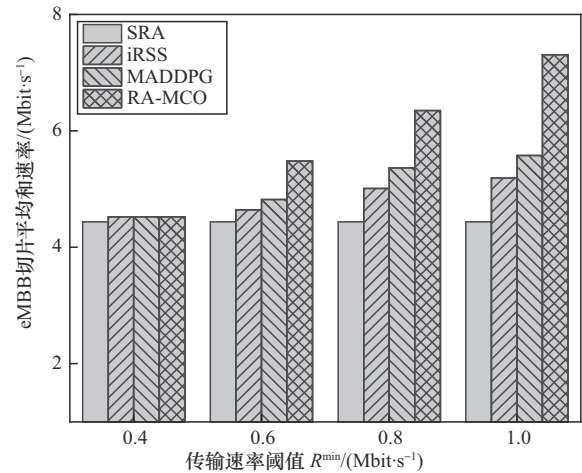


图11 eMBB切片平均和速率与传输速率阈值的关系

图12显示了不同算法在改变PRB总资源量时的eMBB切片平均时延性能。从图12中可以观察到,随着PRB总资源量的增加,各算法的eMBB切片平均时延均呈下降趋势。这是由于可供各切片分配的资源量随PRB总资源量的增加而增加,使用户的传输速率提升,从而减少了平均时延。相较于其他算法,RA-MCO算法在时延降低方面的优势更为显著,其相比SRA、iRSS和MADDPG算法的平均时延分别降低了10.19%、7.86%和4.70%。这是因为该算法在大时间尺度以保障平均时延为约束进行切片资源配置。在小时间尺度切片智能体能够动态调整各切片的资源分配,并且当切片资源不够时,共享资源也参与分配,从而提高系统总传输速率且降低排队时延,以实现系统效用的最大化。相比之下,SRA算法虽然同样表现出用户平均时延的下降趋势,但该算法缺乏灵活的资源分配机制,限制了平均时延性能的进一步提升。iRSS算法的大时间尺度资源需求预测和小时间尺度资源分配都没有考虑时延约束,使其平均时延性能始终高于MADDPG和RA-MCO算法。MADDPG算法由于不存在共享资源,业务动态变化时的资源调整只能等后续PW进行,虽然后续可通过资源重配置改善平均时延,但整体平均时延性能不如RA-MCO算法。

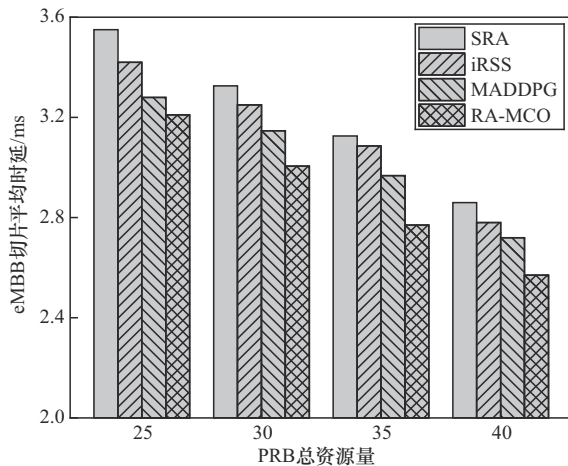


图12 eMBB切片平均时延与PRB总资源量的关系

5 结束语

本文针对6G移动通信场景中无蜂窝MIMO系统,同时考虑了eMBB和uRLLC两类业务,将网络切片无线资源分配问题转换为大时间尺度的周期性切片资源重配置问题和小时间尺度的切片资源动态分配问题。为解决上述问题,在大时间尺度构建切片资源需求的预测问题,通过LSTM实现各切片的资源需求预测。在小时间尺度构建一个部分可观的MDP并转换为多智能体协作问题,通过引入共享资源智能体,结合MAPPO算法进行面向用户的资源自适应分配。从理论上分析了算法的时间复杂度,并通过仿真验证了本文算法的性能。仿真结果表明,本文算法通过切片周期性的重配置以及多智能体之间的协作,能够实现切片资源需求的准确预测,提高了系统传输速率,降低用户平均时延和业务阻塞率,从而提升了系统效用,有效解决了业务动态变化时用户服务质量难以保障的问题。

参考文献:

[1] ALSABAH M, NASER M A, MAHMMOD B M, et al. 6G wireless communications networks: a comprehensive survey[J]. IEEE Access, 2021, 9: 148191-148243.

[2] NGO H Q, ASHIKHMIN A, YANG H, et al. Cell-free massive MIMO versus small cells[J]. IEEE Transactions on Wireless Communications, 2017, 16(3): 1834-1850.

[3] BUZZI S, D'ANDREA C. Cell-free massive MIMO: user-centric approach[J]. IEEE Wireless Communications Letters, 2017, 6(6): 706-709.

[4] NGO H Q, TRAN L N, DUONG T Q, et al. On the total energy efficiency of cell-free massive MIMO[J]. IEEE Transactions on Green

Communications and Networking, 2018, 2(1): 25-39.

[5] AKYILDIZ I F, KAK A, NIE S. 6G and beyond: the future of wireless communications systems[J]. IEEE Access, 2020, 8: 133995-134030.

[6] WANG Z Y, WEI Y F, YU F R, et al. Utility optimization for resource allocation in edge network slicing using DRL[C]//Proceedings of the GLOBECOM 2020-2020 IEEE Global Communications Conference. Piscataway: IEEE Press, 2020: 1-6.

[7] 孙君, 雷振宇. 基于差异性隔离和复用的网络切片无线资源分配方案[J]. 通信学报, 2025, 46(3): 109-121.

SUN J, AI Z Y. Wireless resource allocation scheme for network slicing based on differentiated isolation and multiplexing[J]. Journal on Communications, 2025, 46(3): 109-121.

[8] HALABIAN H. Distributed resource allocation optimization in 5G virtualized networks[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(3): 627-642.

[9] WEN R H, FENG G, TANG J H, et al. On robustness of network slicing for next-generation mobile networks[J]. IEEE Transactions on Communications, 2019, 67(1): 430-444.

[10] WU S C, LIU L Y, ZHANG W B, et al. Revenue-maximizing resource allocation for multitenant cell-free massive MIMO networks[J]. IEEE Systems Journal, 2022, 16(2): 3410-3421.

[11] ZAPPONE A, RENZO M D, DEBBAH M. Wireless networks design in the era of deep learning: model-based, AI-based, or both?[J]. IEEE Transactions on Communications, 2019, 67(10): 7331-7376.

[12] KAMEL M I, LE L B, GIRARD A. LTE wireless network virtualization: dynamic slicing via flexible scheduling[C]//Proceedings of the 2014 IEEE 80th Vehicular Technology Conference (VTC Fall). Piscataway: IEEE Press, 2014: 1-5.

[13] 元伟敬, 宋清洋, 郭磊. 面向软件定义多模态车联网的双时间尺度RAN切片资源分配[J]. 通信学报, 2022, 43(4): 60-70.

QI W J, SONG Q Y, GUO L. Dual time scale resource allocation for RAN slicing in software-defined oriented polymorphic IoV[J]. Journal on Communications, 2022, 43(4): 60-70.

[14] CHEN M Z, CHALLITA U, SAAD W, et al. Artificial neural networks-based machine learning for wireless networks: a tutorial[J]. IEEE Communications Surveys & Tutorials, 2019, 21(4): 3039-3071.

[15] 张海君, 陈安琪, 李亚博, 等. 6G移动网络关键技术[J]. 通信学报, 2022, 43(7): 189-202.

ZHANG H J, CHEN A Q, LI Y B, et al. Key technologies of 6G mobile network[J]. Journal on Communications, 2022, 43(7): 189-202.

[16] LI R P, ZHAO Z F, SUN Q, et al. Deep reinforcement learning for resource management in network slicing[J]. IEEE Access, 2018, 6: 74429-74441.

[17] QI C, HUA Y X, LI R P, et al. Deep reinforcement learning with discrete normalized advantage functions for resource management in network slicing[J]. IEEE Communications Letters, 2019, 23(8): 1337-1341.

[18] HUIN N, MEDAGLIANI P, MARTIN S, et al. Hard-isolation for network slicing[C]//Proceedings of the IEEE INFOCOM 2019-IEEE Con-

- ference on Computer Communications Workshops (INFOCOM WKSHPs). Piscataway: IEEE Press, 2019: 955-956.
- [19] HUA Y X, LI R P, ZHAO Z F, et al. GAN-powered deep distributional reinforcement learning for resource management in network slicing[J]. IEEE Journal on Selected Areas in Communications, 2020, 38(2): 334-349.
- [20] ZHANG H, PAN G J, XU S G, et al. A hard and soft hybrid slicing framework for service level agreement guarantee via deep reinforcement learning[C]//Proceedings of the 2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring). Piscataway: IEEE Press, 2022: 1-5.
- [21] LIU Y N, WANG X B, BOUDREAU G, et al. Deep learning based hot-spot prediction and beam management for adaptive virtual small cell in 5G networks[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2020, 4(1): 83-94.
- [22] YAN M, FENG G, ZHOU J H, et al. Intelligent resource scheduling for 5G radio access network slicing[J]. IEEE Transactions on Vehicular Technology, 2019, 68(8): 7691-7703.
- [23] AZIMI Y, YOUSEFIS, KALBKHANI H, et al. Energy-efficient deep reinforcement learning assisted resource allocation for 5G-RAN slicing[J]. IEEE Transactions on Vehicular Technology, 2022, 71(1): 856-871.
- [24] MEI J, WANG X B, ZHENG K, et al. Intelligent radio access network slicing for service provisioning in 6G: a hierarchical deep reinforcement learning approach[J]. IEEE Transactions on Communications, 2021, 69(9): 6063-6078.
- [25] LI X H, JIAO K J, CHEN X Y, et al. Demand-oriented fog-RAN slicing with self-adaptation via deep reinforcement learning[J]. IEEE Transactions on Vehicular Technology, 2023, 72(11): 14704-14716.
- [26] YE F, LI J M, ZHU P C, et al. Intelligent hierarchical NOMA-based network slicing in cell-free RAN for 6G systems[J]. IEEE Transactions on Wireless Communications, 2024, 23(5): 4724-4737.
- [27] YE F, WANG J, LI J M, et al. Intelligent hierarchical network slicing based on dynamic multi-connectivity in cell-free distributed massive MIMO systems[J]. IEEE Transactions on Vehicular Technology, 2023, 72(9): 11855-11870.
- [28] WANG D M, YOU X H, HUANG Y M, et al. Full-spectrum cell-free RAN for 6G systems: system design and experimental results[J]. Science China Information Sciences, 2023, 66(3): 130305.
- [29] WANG J Y, DAI L, YANG L, et al. Clustered cell-free networking: a graph partitioning approach[J]. IEEE Transactions on Wireless Communications, 2023, 22(8): 5349-5364.
- [30] BUZZI S, D'ANDREA C, ZAPPONE A, et al. User-centric 5G cellular networks: resource allocation and comparison with the cell-free massive MIMO approach[J]. IEEE Transactions on Wireless Communications, 2020, 19(2): 1250-1264.
- [31] HOSSAIN A R, ANSARI N. Priority-based downlink wireless resource provisioning for radio access network slicing[J]. IEEE Transactions on Vehicular Technology, 2021, 70(9): 9273-9281.
- [32] CAO J, WANG D M, LI J M, et al. Uplink spectral efficiency analysis of multi-cell multi-user massive MIMO over correlated Ricean channel[J]. Science China Information Sciences, 2018, 61(8): 082305.
- [33] BJÖRNSON E, SANGUINETTI L. Scalable cell-free massive MIMO systems[J]. IEEE Transactions on Communications, 2020, 68(7): 4247-4261.
- [34] YANG H J, ZHENG K, ZHANG K, et al. Ultra-reliable and low-latency communications for connected vehicles: challenges and solutions[J]. IEEE Network, 2020, 34(3): 92-100.
- [35] SONG F, LI J, MA C, et al. Dynamic virtual resource allocation for 5G and beyond network slicing[J]. IEEE Open Journal of Vehicular Technology, 2020, 1: 215-226.
- [36] NEELY M J. Stochastic network optimization with application to communication and queueing systems[J]. Synthesis Lectures on Communication Networks, 2010, 3(1): 1-199.
- [37] BERTSEKAS D, GALLAGER R. Data networks (2nd ed.)[M]. New York, Prentice-Hall, 1992.
- [38] YU A, YANG H, NGUYEN K K, et al. Burst traffic scheduling for hybrid E/O switching DCN: an error feedback spiking neural network approach[J]. IEEE Transactions on Network and Service Management, 2021, 18(1): 882-893.
- [39] SAK H, SENIOR A, BEAUFAYS F. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition[J]. Computer Science, 2014: 338-342.
- [40] HOU W J, WEN H, SONG H H, et al. Multi-agent deep reinforcement learning for task offloading and resource allocation in cybertwin based networks[J]. IEEE Internet of Things Journal, 2021, 8(22): 16256-16268.
- [41] ZENG W B, HE Y G, LI B, et al. Pilot assignment for cell free massive MIMO systems using a weighted graphic framework[J]. IEEE Transactions on Vehicular Technology, 2021, 70(6): 6190-6194.
- [42] KIM J, KIM S, BANG J, et al. Adaptive mode selection in D2D communications considering the bursty traffic model[J]. IEEE Communications Letters, 2016, 20(4): 712-715.

[作者简介]



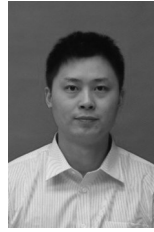
夏玮玮 (1975-), 女, 江苏句容人, 东南大学副研究员、博士生导师, 主要研究方向为无线网络资源管理、边缘计算、泛在网络与短距离无线通信等。



王博业 (2001-), 男, 江苏南京人, 东南大学硕士生, 主要研究方向为无线网络资源管理、6G通信等。



夏雅星 (2002-), 女, 江苏宿迁人, 东南大学硕士生, 主要研究方向为无线资源调度优化、6G工业物联网通信等。



景栋盛 (1981-), 男, 江苏苏州人, 国网苏州供电公司高级工程师, 主要研究方向为电力系统数字化、网络安全、物联网技术、电力通信技术。



缪巍巍 (1968-), 男, 江苏如皋人, 国网江苏省电力有限公司正高级工程师, 主要研究方向为电力信息通信网络、电力物联网、电力无线专网等。



燕锋 (1983-), 男, 湖北天门人, 东南大学副教授、博士生导师, 主要研究方向为无人机自组网、卫星互联网、无线传感器网络等。



汪大洋 (1987-), 男, 江苏南京人, 国网江苏省电力有限公司高级工程师, 主要研究方向为电力通信等。



沈连丰 (1952-), 男, 江苏邳州人, 东南大学教授、博士生导师, 主要研究方向为宽带移动通信、短距离无线通信、泛在网络等。