引用本文：王蕴，林霄，楼芝兰，等. 面向边缘光算力网络的上行链路资源协同调度算法[J]. 光通信技术，2024，48(3)：45-51.

面向边缘光算力网络的上行链路资源协同调度算法

王 蕴[1]，林 霄[1*]，楼芝兰[2]，李 军[3]，孙卫强[4]

（1.福州大学 物理与信息工程学院，福州 350116；2.浙江财经大学 数据科学学院，杭州 310018；3.苏州大学 江苏省新型光纤技术与通信网络工程研究中心，江苏 苏州 215006；4.上海交通大学 区域光纤通信网与新型光通信系统国家重点实验室，上海 200240）

摘要：为满足冷、热业务实时、高效的算力调度需求，提出一种基于自适应噪声完全集合经验模态分解（CEEMDAN）与时间卷积网络（TCN）的算力负载预测模型（简称C-TCN模型），并设计了基于C-TCN与Q学习的资源协同调度算法（CTQ算法），利用C-TCN模型提前感知下一时刻负载变化，通过Q学习协同调度波长与存储资源，寻找最佳波长划分与边缘存储分配方案。实验结果表明：CTQ算法的调度性能不仅优于现有调度算法，能满足冷、热业务调度性能要求，而且还能提高波长利用率。

关键词：边缘光算力网络；算力调度；数据传输；资源调度；网络优化

中图分类号：TN91　**文献标志码**：A　**文章编号**：1002-5561(2024)03-0045-07

DOI：10.13921/j.cnki.issn1002-5561.2024.03.009　**开放科学（资源服务）标识码（OSID）**：

Uplink resource coordinated scheduling algorithm for edge-oriented optical computing power networks

WANG Yun[1], LIN Xiao[1*], LOU Zhilan[2], LI Jun[3], SUN Weiqiang[4]

(1. College of Physics and Information Engineering, Fuzhou University, Fuzhou 350116, China; 2. School of Data Sciences, Zhejiang University of Finance & Economics, Hangzhou 350018, China; 3. Jiangsu New Optical Fiber Technology and Communication Network Engineering Research Center, Soochow University, Suzhou Jiangsu 215006, China; 4. State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract: In order to meet the real-time and efficient computing power scheduling requirements of hot and cold services, a computational load prediction model (abbreviated as C-TCN model) based on adaptive noise complete set empirical mode decomposition (CEEMDAN) and time convolutional network (TCN) is proposed, and a resource cooperative scheduling algorithm (CTQ algorithm) based on C-TCN and Q learning is designed. The C-TCN model is used to sense the load change at the next time in advance, and the optimal wavelength partitioning and edge storage allocation scheme is found through Q learning. The experimental results show that the CTQ algorithm not only has better scheduling performance than the existing scheduling algorithms, but also can meet the requirements of hot and cold service scheduling performance, and improve the wavelength utilization rate.

Key words: edge optical computing power network, computing power scheduling, data transfer, resource scheduling, network optimization

0 引言

近年来，大语言模型、生成式人工智能（AI）等新一代AI技术快速发展，取得了诸多令人瞩目的成果[1]。AI技术的蓬勃发展致使海量亟待训练与推理的数据在网络边缘侧产生[2]。然而，终端设备、边缘节点通常计算资源有限，无法满足AI“最后一千米”的算力需求[3]。为此，国家六部门联合印发文件[4]明确提出促进边缘算力协同部署，强化算力接入网络能力，通过调度各区域节点之间的计算资源，满足日益增长的边缘算力需求。

与电网中电力调度不同，算力无法在物理维度被

收稿日期：2024-02-06。

基金项目：国家自然科学基金青年项目（批准号：61901118、12001483）资助；上海交通大学“区域光纤通信网与新型光通信系统国家重点实验室”开放基金（批准号：2023GZKF020）资助；江苏省新型光纤技术与通信网络工程研究中心开放研究课题（批准号：SDGC2232）资助。

作者简介：王蕴（1998—），男，硕士研究生，本科毕业于福州大学物联网工程专业，现就读于福州大学物理与信息工程学院新一代电子信息技术（含量子技术等）专业，主要从事算力网络控制、智能光网络优化方面的研究工作。

*** 通信作者**：林霄（1988—），男，博士，副教授，主要从事算力网络、人工智能、大数据传输、智能光网络等相关领域的理论及应用研究工作。

调度至用户。因此，算力调度的核心是通过网络将用户数据传输至目标节点进行计算。光网络因其高带宽、低时延、高可靠性及确定性等特点[5]，使其成为边缘算力网络的承载网，即边缘光算力网络（EO-CPN），为算力调度提供高效且可靠的传输通道。然而，EO-CPN 效能的发挥取决于其根据算力业务特点，对光网络资源进行高效调度的能力。

现有的调度算法主要研究静态网络场景，即业务到达网络情况提前给定或者已知，并将网络资源调度问题建模为静态优化问题[6-8]。在实际网络中，算力业务到达情况是时变的，因此现有静态优化调度算法难以满足算力业务的实时调度需求。此外，算力业务通常可分为冷业务和热业务两大类[9]。其中，热业务以在线游戏、AI 推理等为代表，这类业务对传输延迟的要求较高，因此需要立即调度资源以满足其算力需求；而冷业务则以大数据分析、AI 模型训练等为代表，这类业务在遭遇网络拥塞或计算高峰时，可容忍一定延迟，因此可以通过延迟调度来“错峰”满足算力需求。然而，大部分现有调度算法只研究单一类型业务，缺乏对算力业务冷热属性的区分，难以高效利用宝贵的网络资源[6-8]。其原因为：一方面，若所有业务按照高峰期流量需求进行网络资源配置，可能导致资源利用率低下、算力业务调度成本过高；另一方面，若所有业务均按冷业务要求调度，将造成网络资源配置不足，导致大量热业务将因长时间等待有限资源而无法及时调度，最终造成阻塞。文献[10]考虑了边缘计算中同时存在的延时敏感型与延时容忍型任务，并设计了资源划分、抢占和缓存机制，这些机制协同满足 2 种类型边缘计算任务的调度需求。然而，多种调度机制的引入使得调度问题复杂度高，而传统优化算法求解过程复杂，因此难以实现实时调度。

本文将自适应噪声完全集合经验模态分解（CEEMDAN）与时间卷积网络模型（TCN）相结合，提出一种算力负载预测模型（下文简称 C-TCN 模型），以准确预测下一时刻算力负载的动态变化。在此基础上，设计基于 C-TCN 与 Q 学习的资源协同调度算法（下文简称 CTQ 算法），满足冷热业务调度性能要求的同时提高资源利用效率。

1 算力调度系统

1.1 EO-CPN 模型

EO-CPN 模型如图 1 所示，远端算力节点就近部署于算力需求侧，通过光纤不仅与多个算力客户网相连，还与计算资源更为丰富的城域算力节点相连。算力节点包含光交换器、IP 路由器和算力服务器。软件定义网络（SDN）控制器负责对链路上波长资源和算力服务器上存储资源进行集中式管理和调度。EO-CPN 模型共有 3 种典型算力客户网：工业互联网、移动互联网和企业网。其中，工业互联网中的生产监测、机器人控制等应用，以及移动互联网中在线游戏、视频会

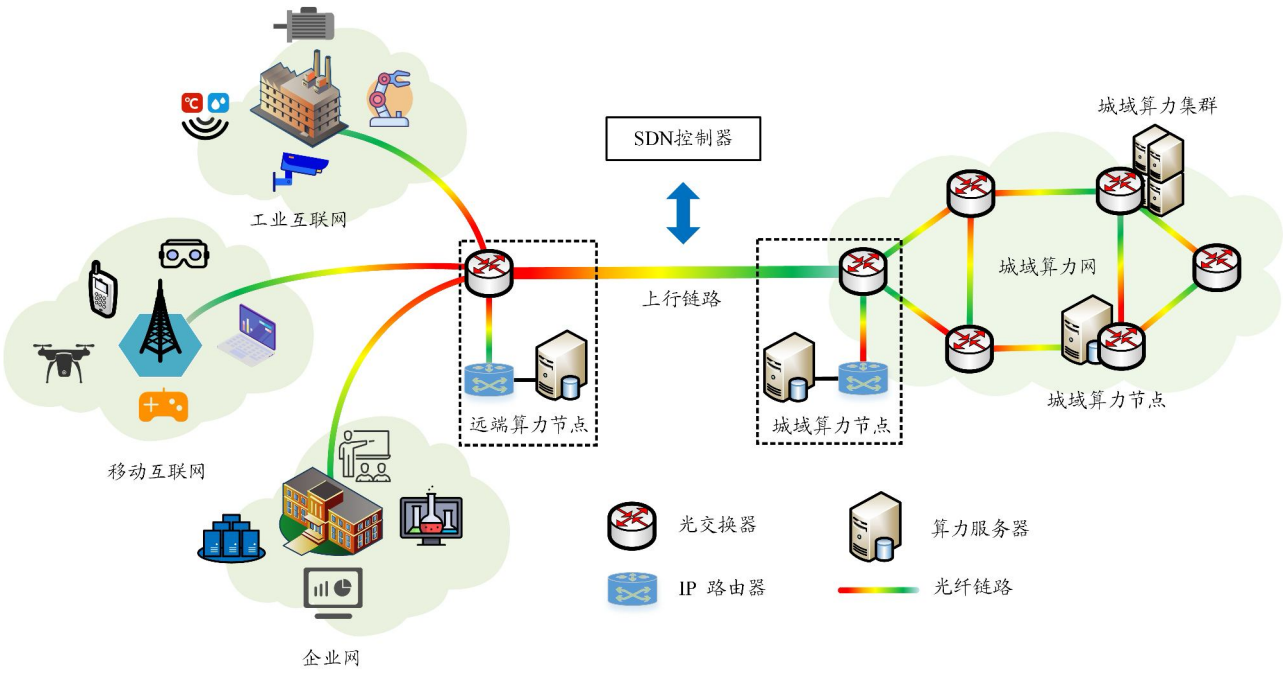


图 1 EO-CPN 模型

议等应用对算力需求均以热业务为主；企业网中数据分析、迁移备份等应用对算力需求均以冷业务为主。通常远端算力节点足以服务来自客户网的算力负载。然而，当算力负载处于使用高峰时段，客户侧产生的冷热业务计算请求超出远端算力节点的服务能力，此时可通过 SDN 控制器将算力业务的数据经上行光纤链路发送至城域算力节点进行计算，从而满足算力业务的调度需求。本文假设城域算力节点间距离有限，借助光网络提供的低时延传输通道，冷热业务在远端与城域算力节点间传输延迟可以得到有效保障。

假设需要从远端节点传输至城域节点的算力业务的到达服从泊松分布，其到达率为 λ 个业务/秒，业务待传输数据量服从负指数分布，其均值为 F Gb。假设每个业务使用一个波长传输数据，数据速率为 c Gb/s，则算力业务的服务率 $\mu=c/F$，算力负载 $\rho=\lambda/\mu$。热业务在所有算力业务负载中占比为 δ。远端节点传输至城域节点光纤链路的波长资源中可用于算力调度的波长总数为 w_{total}。与文献[10]相同，本文将 w_{total} 划分为 w_{hot}、w_{cold} 和 w_{share} 3 个区域，其中 w_{hot} 为热业务传输的波长数，w_{cold} 为冷业务传输的波长数，w_{share} 为冷热业务共用波长数。当新到达的热业务发现波长资源不足时，允许其在 w_{share} 中抢占正在传输的冷业务波长。若 w_{share} 中无足够波长可供抢占，则阻塞热业务。当新到达的冷业务发现波长资源不足时，允许其暂时缓存于远端算力节点，等待 w_{cold} 或 w_{share} 有空闲波长时再进行传输。远端算力节点中可用于缓存冷业务的边缘存储资源为 S_{edge}。若 S_{edge} 没有足够的存储空间用于缓存，则阻塞新到达的冷业务。令 B_{cold}、B_{hot}、P_{hot} 和 U 分别为冷业务阻塞率、热业务阻塞率、热业务抢占率和波长资源利用率，并分别用于衡量调度系统中冷热业务的调度性能和波长资源的使用情况。B_{cold}、B_{hot}、P_{hot} 和 U 均为 w_{hot}、w_{cold}、w_{share}、S_{edge}、ρ 和 δ 的函数。

1.2 优化模型

为满足 EO-CPN 场景中冷热业务调度需求，现将波长与存储资源调度问题建模为优化问题。一方面，增大 w_{hot} 和 w_{cold} 以有效降低 B_{hot} 和 B_{cold}，但这会降低 U，大量宝贵的波长资源可能被闲置，造成调度成本高昂；另一方面，增大 w_{share} 和 S_{edge} 可以提高 U，但会导致 P_{hot} 升高，从而加剧冷业务被抢占的情况。为此，本文为波长划分与边缘存储资源分配问题建立优化模型，以实现业务性能与资源利用率之间的平衡。

首先，通过调整 w_{hot}、w_{share}、w_{cold} 和 S_{edge}，在最小化 B_{hot}、B_{cold}、P_{hot} 的同时，最大化 U。然后，将阻塞率和抢占率映射至以 10 为底的对数域，提高目标函数对极小性能数值的灵敏度。此外，B_{hot}、B_{cold}、P_{hot} 的对数值之和为负数，而 U 大于 0，因此最大化目标函数即可得到最小化的 B_{hot}、B_{cold}、P_{hot} 和最大化 U。ζ、η 和 θ 为大于等于 0 的权重因子，可根据调度需求进行调整。本文目标函数如式(1)所示。

$$\max\ r_0=U\times\left|\zeta\times\lg B_{\text{hot}}+\eta\times\lg B_{\text{cold}}+\theta\times\lg P_{\text{hot}}\right| \tag{1}$$

各调度性能约束条件如下：

$$B_{\text{cold}}^{\text{Min}}\leqslant B_{\text{cold}}\leqslant B_{\text{cold}}^{\text{Max}} \tag{2}$$

$$B_{\text{hot}}^{\text{Min}}\leqslant B_{\text{hot}}\leqslant B_{\text{hot}}^{\text{Max}} \tag{3}$$

$$P_{\text{hot}}^{\text{Min}}\leqslant P_{\text{hot}}\leqslant P_{\text{hot}}^{\text{Max}} \tag{4}$$

$$U^{\text{Min}}\leqslant U \tag{5}$$

其中，$B_{\text{cold}}^{\text{Max}}$、$B_{\text{hot}}^{\text{Max}}$ 分别表示冷、热业务可接受阻塞率上界，$B_{\text{cold}}^{\text{Min}}$、$B_{\text{hot}}^{\text{Min}}$ 分别表示冷、热业务可接受阻塞率下界，$P_{\text{hot}}^{\text{Max}}$、$P_{\text{hot}}^{\text{Min}}$ 分别表示热业务抢占率可接受的上、下界。

式(2)、式(3)和式(4)分别保证冷热业务的阻塞率和热业务抢占率不会过低或过高，而式(5)则用来确保波长资源利用率不会过低。

资源约束条件如下：

$$w_{\text{total}}=w_{\text{hot}}+w_{\text{share}}+w_{\text{cold}} \tag{6}$$

$$0\leqslant w_{\text{total}}\leqslant W_{\text{total}}^{\text{Max}} \tag{7}$$

$$0\leqslant S_{\text{edge}}\leqslant S_{\text{edge}}^{\text{Max}} \tag{8}$$

式(6)与式(7)确保在调度过程中，所使用的波长资源不超过可用于算力调度的最大波长资源 $W_{\text{total}}^{\text{Max}}$；式(8)确保用于缓存冷业务的边缘存储资源不超过远端算力节点中算力服务器可用于业务缓存的存储总量 $S_{\text{edge}}^{\text{Max}}$。

2 算力负载预测 C-TCN 模型

2.1 基于 CEEMDAN 的算力负载时间序列数据分解与重构

算力负载在以天、周为单位的宏观时间尺度上呈现周期性变化特征，而在以分钟、小时为单位的微观时间尺度上呈现非线性、非平稳性变化特征。现有预测模型，如差分整合移动平均自回归模型(ARIMA)[11]和门控循环单元(GRU)[12]等，虽然这些模型在大时间尺度上取得较好的预测效果，但是在更精细时间粒度上预测精确度不足。为了克服上述问题，本文采用 CEEMDAN[13]对原始算力负载时间序列进行预处理，以

滑动窗口方式，将窗口内的时间序列数据分解为一系列本征模态函数(IMF)分量和一个残差(RES)分量，以便捕捉不同时间尺度下序列变化特征。为了降低模型训练负担的同时保证模型训练效果，本文保留 IMF 中高频分量 IMF1 和 IMF2，并将其余低频分量和 RES 再次重构为新序列用于 TCN 模型的训练。同时，使用互联网流量数据集[14]模拟算力业务到达率 λ 随时间变化情况。假设 μ 不随时间改变，因此对 λ 的预测等价于对算力负载 ρ 的预测。

2.2 C-TCN 模型结构

与传统循环神经网络相比，TCN 模型在处理时间序列方面具有训练速度快、记忆性好和感受野大等特点[15]。因此，本文使用 3 个 TCN 模块对 IMF1、IMF2 和重构序列分别进行学习，再将 3 个 TCN 模块的输出都送入全连接模块融合并输出，从而预测下一个时刻的算力业务到达率 $\hat{\lambda}$。其中，TCN 模块的特征选取窗口长度为 5，隐藏层数为 4，隐藏层卷积核数量为 64，卷积核长度为 6，卷积核膨胀因子为 2，输出量为 1，选取 ReLU 作为激活函数、平均绝对误差(MAE)作为损失函数。全连接模块分 3 层，其中输入层包含 3 个神经元，输出层包含 1 个神经元，隐藏层包含 64 个神经元，选取 ReLU 作为激活函数、均方误差(MSE)作为损失函数。

在训练过程中，本文选择 85%数据集作为训练集，其余15%作为测试集。使用 20%的训练样本作为训练时的验证集。选择 Adam 作为优化器，其初始学习率为 0.000 8。此外，为避免训练过程中的过拟合问题，设置早停机制，若连续 20 轮训练的验证损失值没有下降，则终止训练。

3 上行链路资源协同调度算法

多业务、多机制、多优化目标的共存导致优化模型过于复杂，传统优化算法难以求解，从而无法满足实时调度需求。为此，本文利用 Q 学习在实用性、收敛性和无模型学习方面的优势，设计了 CTQ 算法，以实现各调度性能间的平衡。该算法通过线下训练 Q 表，线上决策方式实现调度。在线上决策时，调度系统只需通过查询 Q 表即可找到最佳波长划分与存储分配方案。

Q 学习主要由智能体(Agent)、状态 s、动作 a、奖励 r 和 Q 表组成。在学习之前，创建初始值为 0 的 Q 表，其行和列分别代表动作和状态。每次迭代时，Agent 都会根据当前状态 s 和指定策略选择动作 a 执行。在动作执行后，状态从 s 转移到 s'，Agent 从环境的反馈中得到奖励 r 并更新 Q 表。

Q 表更新公式如下

$$Q(s,a)\leftarrow Q(s,a)+\alpha[r+\gamma\max Q(s',a')-Q(s,a)] \tag{9}$$

其中，α 为学习率($0\leqslant\alpha\leqslant1$)，$\gamma$ 为折扣因子($0\leqslant\gamma\leqslant1$)。

根据 1.2 节中的优化模型，本文对状态空间、动作空间和奖励函数设计如下：

1)状态空间：Agent 需要获知当前波长划分、边缘存储分配和算力业务到达情况，以此来开展资源调度，因此状态空间设计如下：

$$S=\{s\mid s=(w_{\text{total}},w_{\text{hot}},w_{\text{share}},S_{\text{edge}},\lambda)\} \tag{10}$$

2)动作空间：动作表示针对当前波长划分与边缘存储分配的改变方式。假设 Δw 为调度系统中波长划分改变的最小量，ΔS_{edge} 为边缘存储改变的最小量，则 w_{total}、w_{hot}、w_{cold} 与 S_{edge} 所对应的动作集可分别表示为 $a_1=\{-\Delta w,0,\Delta w\}$、$a_2=\{-\Delta w,0,+\Delta w\}$、$a_3=\{-\Delta w,0,\Delta w\}$ 和 $a_4=\{-\Delta S_{\text{edge}},0,+\Delta S_{\text{edge}}\}$。其中，$-\Delta w$ 与 $+\Delta w$ 分别表示减少或增加 Δw 数量波长的操作，$-\Delta S_{\text{edge}}$ 与 $+\Delta S_{\text{edge}}$ 分别表示对减少或增加 ΔS_{edge} 数量存储的操作，0 表示保持现有资源，不做任何操作。本文联合 a_1、a_2、a_3、a_4 作为最终动作，动作空间设计如下：

$$A=\{a\mid a=(a_1,a_2,a_3,a_4)\} \tag{11}$$

3) 奖励函数：Agent 学习的首要目标是满足所有约束条件，在此前提下最大化目标函数。假设当前状态下已满足的约束条件数量为 n，约束条件总数为 N，式(1)计算结果为 r_0。在训练过程中，当 $n<N$ 时，Agent 的目标是通过选取合适的动作 a，使得 n 增大；当 $n=N$ 时，Agent 的目标是通过选取合适的动作 a，最大化目标函数。故奖励函数设计如下：

$$r=\begin{cases}x, n<N\\ k\times r_0, n=N\end{cases} \tag{12}$$

其中，x 为约束条件不能完全满足(即 $n<N$)时系统所给予的奖励；k 为所有约束条件均能满足 (即 $n=N$ 时)根据 r_0 变化情况而调整的权重奖励因子。具体而言，当 $n<N$ 时，在动作 a 执行完成后，若 n 增大，则 x 设为 10；若 n 保持不变，则 x 设为 −1；若 n 减小，则 x 设为 −10。当 $n=N$ 时，在执行完动作 a 后，若 r_0 增大，则 k 设为 2；若 r_0 保持不变，则 k 设为 1；若 r_0 减小，则 k 设为 −4。

为了加快收敛速度，本文还采用动态探索率 ε 更新机制，设置最大值 $\varepsilon_{\max}$ 与最小值 $\varepsilon_{\min}$。在训练初期，为让 Agent 快速探索到足够多的动作和状态，将探索率

设为 ε_{max}。随着训练的推进，Agent 所学到的策略逐渐稳定，探索率随之降低直至 ε_{min}。CTQ 算法训练流程如算法 1 所示。其中，M 为迭代轮次，T 为数据集[14]中用于训练不同时刻的样本数量。在线操作阶段中，于时刻 t 开始，需按照以下流程逐步执行：1）确定当前状态 s_t，并根据 s_t 使用 C-TCN 模型预测时刻 t+1 到达率；2）根据 s_t 在 Q 表中查找现有最优决策，使得执行动作 a 时获得最大 Q 值。

算法 1　CTQ 算法训练流程

输入：状态空间、动作空间、Q 表、学习率 α、折扣因子 γ 以及 ε-greedy 策略的探索率 ε

输出：Q 表

1）for each episode m=1 to M do

2）　设置初始到达率，以及初始资源划分与边缘存储分配

3）　for each time i=1 to T do

4）　根据当前时刻 t_i 的实际达到率 λ_i、资源划分与边缘存储分配，获得当前状态 s

5）　基于 ε-greedy 策略进行动作选择：生成随机数，若随机数小于 ε，则随机选取可执行动作 a，否则根据 Q 表选择 Q 值最大的可执行动作 a

6）　执行动作 a，观测 EO-CPN 网络仿真环境所反馈的 B_{hot}、B_{cold}、P_{hot} 和 U，从而计算奖励 r

7）　使用 C-TCN 模型预测 t_{i+1} 时刻到达率 $\hat{\lambda}_{i+1}$，从而获取下一个时刻状态 s'

8）　根据下式更新 Q 表：

$$Q(s,a)\leftarrow Q(s,a)+\alpha[r+\gamma \max Q(s',a')-Q(s,a)]$$

9）　更新探索率 ε：$\varepsilon\leftarrow\varepsilon_{min}+(\varepsilon_{max}-\varepsilon_{min})\times\frac{(i+m\times T)}{M\times T}$

10）　end for

11）end for

4 算法仿真结果分析

C-TCN 模型的预测结果不仅用于 CTQ 的训练阶段，还用于执行阶段。因此，C-TCN 模型的预测准确性对调度性能的影响至关重要。本文首先对比现有预测模型与 C-TCN 模型的预测性能，以及现有调度算法与 CTQ 算法的 B_{hot}、B_{cold}、P_{hot}、U 性能。

4.1 算力负载预测模型比较分析

本文采用平均绝对误差（MAE）、平均绝对百分比误差（MAPE）和均方误差（MSE）作为模型预测能力的评估指标，并选取 ARIMA、GRU 和 TCN 模型作为比较对象，设置如下：

1）ARIMA 模型[11]：自回归项数 P、差分阶数 D 和滑动平均项数 Q 分别设置为 0、1、4。

2）GRU 模型[12]：输入层包括 5 个神经元，输出层包括 1 个神经元，模型隐藏层数为 4，隐藏层包含 64 个神经元，选取 ReLU 作为激活函数、MAE 作为损失函数。

3）TCN 模型[15]：TCN 模型参数设置和激活函数、损失函数选择与 2.2 节中 C-TCN 模型所用 TCN 模块结构一致。

预测性能比较结果如表 1 所示。可以看出，C-TCN 模型在 MAE、MAPE、MSE 方面均优于其它预测模型。这利益于 CEEMDAN 对原始到达率序列的预处理及 TCN 模型的训练。与直接使用原始数据集进行学习训练的模型相比，C-TCN 模型可以充分挖掘出原始序列中所隐含的特征信息并进行学习训练，因此其预测性能更好。

表 1　预测性能比较结果

预测模型	MAE	MAPE/%	MSE
ARIMA	7.96×10^{-2}	2.93	1.23×10^{-2}
GRU	7.80×10^{-2}	2.92	1.21×10^{-2}
TCN	7.56×10^{-2}	2.76	1.20×10^{-2}
C-TCN	2.96×10^{-2}	1.07	1.73×10^{-3}

4.2 上行链路资源协同调度算法比较分析

本文遵循 1.1 节网络模型与假设，搭建仿真实验环境，实验参数设置如表 2 所示。在每次实验中，远端算力节点随机收到等待传输到城域网节点的算力业务。根据当前调度算法给出的波长划分与存储分配方案，节点遵循先到先服务的原则处理业务，确保资源传输的顺畅进行。同时，利用互联网流量数据集[14]模拟仿真过程中到达率 λ 随时间的变化情况，以更准确地反映实际网络环境。每次仿真实验产生 10^8 个业务，重复20 次实验后取平均值作为最终结果。为了验证 CTQ 算法的高度性能，本文另选取以下 3 种调度算法进行比较：

1）最优调度算法（简称 Optimal 算法）：该算法假设对算力负载变化能进行完美预测，并采用贪心算法遍历所有波长划分和存储分配方案，直至找到最大化目标函数的方案。

2）基于负载均值的固定调度算法（简称 Fixed-Avg

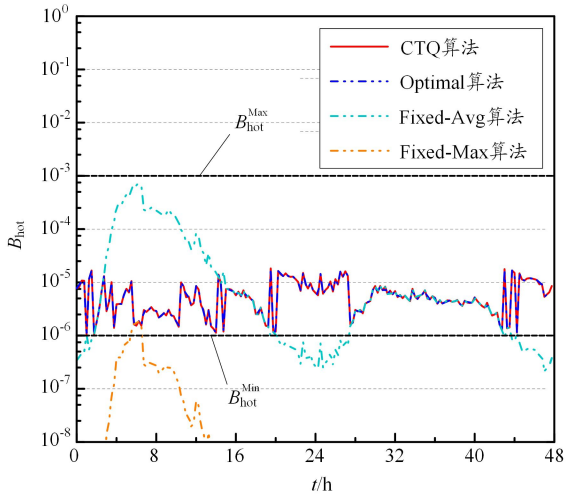
表 2 实验仿真

参数	数值	参数	数值	参数	数值
B_{cold}^{Min}	10^{-6}	B_{cold}^{Max}	10^{-3}	B_{hot}^{Min}	10^{-6}
B_{hot}^{Max}	10^{-3}	P_{hot}^{Min}	10^{-6}	P_{hot}^{Max}	10^{-3}
U^{Min}	0.2	Δw	1	ΔS_{edge}/Gb	1
W_{total}^{Max}	15	S_{edge}^{Max}/Gb	10	λ	[1,6]
δ	0.2	α	0.1	μ	1
ε_{min}	0.01	ε_{max}	0.9	γ	0.9
F/Gb	1	c/Gb/s	1	ρ	[1,6]

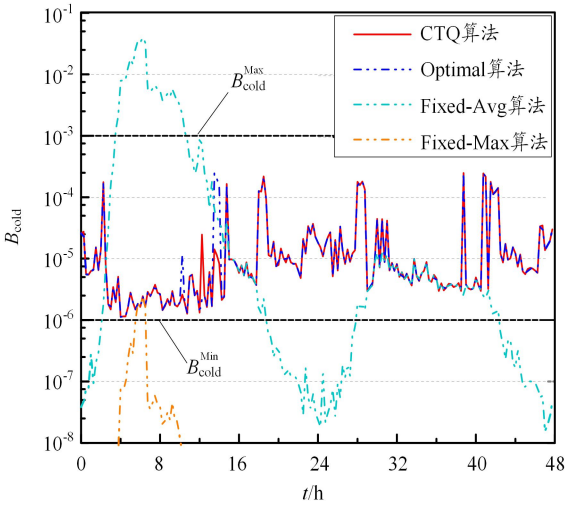
算法)：该算法采用静态波长划分和存储分配方案，不随算力负载而改变。Fixed-Avg 算法选取波长数、边缘存储数与 CTQ 算法在算力负载处于平均水平时的分配方案一致。

3)基于负载峰值的固定调度算法(简称 Fixed-Max 算法)：该算法也采用静态波长划分和存储分配方案，但其选取波长数、边缘存储数与 CTQ 算法在算力负载达峰期所用分配方案一致。

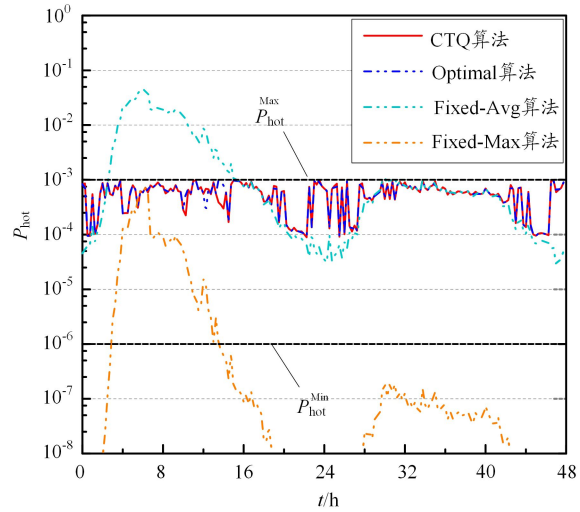
4 种算法的 B_{hot}、B_{cold}、P_{hot}、U 的性能对比如图 2 所示。可以看出，CTQ 算法的调度性能与 Optimal 算法最接近。然而，在实际使用场景中，Optimal 算法完美预测算力负载变化的假设难以成立，且由于采用贪心算法遍历所有方案，导致算法搜索时间长，难以满足 EO-CPN 中算力业务的实时调度需求。CTQ 算法则采用线下训练的方式，在做决策时只需要对 Q 表进行查询即可，决策复杂度低，调度决策效率高。由于 Fixed-Avg



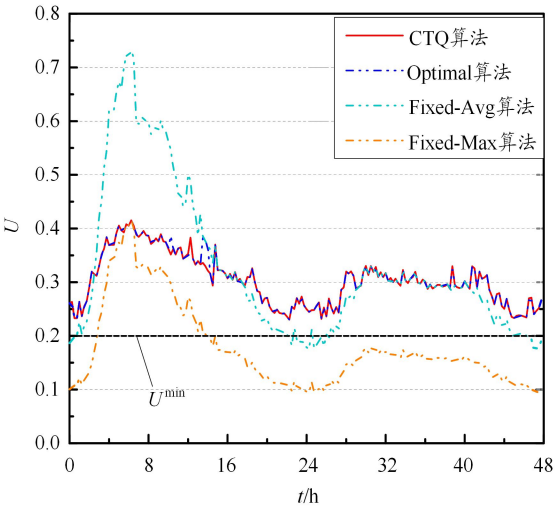
(a) B_{hot}



(b) B_{cold}



(c) P_{hot}



(d) U

图 2 4 种算法调度性能比较

算法采用均值方案，当算力负载出现高峰或者低谷时，可能导致资源配置不足或超额配置，使其调度性能无法满足上界或者下界约束要求。Fixed-Max 算法的峰值方案导致资源超额配置，这使得 Fixed-Max 算法的调度性能在大多数时候都低于性能约束下界，而且波长利用率过低，导致大量宝贵的网络资源浪费。

5 结束语

为了满足 EO-CPN 中冷、热业务的实时算力调度需求，本文首先将 CEEMDAN 与 TCN 模型结合，提出了 C-TCN 模型，用于预测下一时刻算力负载变化。在此基础上，本文将 Q 学习与 C-TCN 模型结合，提出了 CTQ 算法，该算法能够提前感知下一时刻负载变化，预先对波长资源划分与边缘存储资源分配进行协同调度。实验表明，C-TCN 模型的预测精度高于现有预测模型。此外，CTQ 算法在 B_{hot}、B_{cold}、P_{hot}、U 等方面表现出色，不仅优于传统的固定分配算法，还与 Optimal 算法所获得的最优方案最为接近，同时其调度复杂度远低于 Optimal 算法。因此，CTQ 算法能够满足算力业务的实时调度需求。

参考文献：

[1] 杨小康，许岩岩，陈露，等. AI for Science：智能化科学设施变革基础研究[J]. 中国科学院院刊，2024，39(1)：59-69.

[2] 王睿，齐建鹏，陈亮，等. 面向边缘智能的协同推理综述[J]. 计算机研究与发展，2023，60(2)：398-414.

[3] 王晓飞. 智慧边缘计算：万物互联到万物赋能的桥梁[J]. 人民论坛·学术前沿，2020，(9)：6-17，77.

[4] 中华人民共和国中央人民政府. 工业和信息化部等六部门关于印发《算力基础设施高质量发展行动计划》的通知[EB/OL]. (2023-10-08)[2024 -02 -06]. https://www.gov.cn/zhengce/zhengceku/202310/content_6907900.htm.

[5] 逯向军. 基于 OTN 的算力网络承载技术研究[J]. 长江信息通信，2023，36(4)：8-13.

[6] HUANG S, YANG C, YIN S, et al. Latency-aware task peer offloading on overloaded server in multi-access edge computing system interconnected by metro optical networks [J]. Journal of Lightwave Technology, 2020, 38(21): 5949-5961.

[7] LI Y, ZENG Z, LI J, et al. Distributed model training based on data parallelism in edge computing-enabled elastic optical networks [J]. IEEE Communications Letters, 2020, 25(4): 1241-1244.

[8] YI C, CAI J, SU Z. A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications [J]. IEEE Transactions on Mobile Computing, 2019, 19(1): 29-43.

[9] 贾庆民，胡玉姣，张华宇，等. 确定性算力网络研究[J]. 通信学报，2022，43(10)：55-64.

[10] LIN X, LI Y, SHAO J, et al. Storage-assisted optical upstream transport scheme for task offloading in multi-access edge computing [J]. Journal of Optical Communications and Networking, 2022, 14(3): 140-152.

[11] 汪尧，黄宁，武润升，等. 基于改进自回归差分移动平均模型的网络流量预测[J]. 通信技术，2021，54(12)：2626-2631.

[12] 徐海兵，郭久明. 基于双向 GRU 模型的网络流量预测的研究[J]. 电子技术应用，2022，48(2)：19-22，27.

[13] ZHANG S, LIU H, HU M, et al. An adaptive CEEMDAN thresholding denoising method optimized by nonlocal means algorithm[J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69(9): 6891-6903.

[14] JESSICA M. internet-traffic-stats-project[DB/OL]. (2017-07-25)[2024-02-06]. https://github.com/rankinjl/internet-traffic-stats-project.

[15] HEWAGE P, BEHERA A, TROVATI M, et al. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station [J]. Soft Computing, 2020, 24: 16453-16482.