

融合机器学习预测和水波优化算法求解银行在线客服调度问题

卢雪琴, 谢歛铖, 唐燕, 陈世昆, 刘仰光

引用本文

卢雪琴, 谢歛铖, 唐燕, 陈世昆, 刘仰光. [融合机器学习预测和水波优化算法求解银行在线客服调度问题](#)[J]. 计算机科学, 2025, 52(10): 33-49.

LU Xueqin, XIE Xichen, TANG Yan, CHEN Shikun, LIU Yangguang. [Integration of Machine Learning Prediction and Water Wave Optimization for Online Customer Service Representatives Scheduling in Bank Contact Centers](#) [J]. Computer Science, 2025, 52(10): 33-49.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于深度强化学习的安全感知服务功能链部署方法](#)

Security-aware Service Function Chain Deployment Method Based on Deep Reinforcement Learning
计算机科学, 2025, 52(10): 404-411. <https://doi.org/10.11896/jsjcx.240800015>

[面向人机协作的智能体训练方法研究综述](#)

Review of Research on Agent Training Methods Toward Human-Agent Collaboration
计算机科学, 2025, 52(10): 176-189. <https://doi.org/10.11896/jsjcx.241000047>

[基于多智能体深度强化学习的光储充电站动态定价及能源调度策略](#)

Dynamic Pricing and Energy Scheduling Strategy for Photovoltaic Storage Charging Stations Based on Multi-agent Deep Reinforcement Learning
计算机科学, 2025, 52(9): 337-345. <https://doi.org/10.11896/jsjcx.240700197>

[基于图注意力的分组多智能体强化学习方法](#)

Graph Attention-based Grouped Multi-agent Reinforcement Learning Method
计算机科学, 2025, 52(9): 330-336. <https://doi.org/10.11896/jsjcx.240700107>

[基于拥塞感知和缓存通信的多智能体路径规划](#)

Congestion-aware and Cached Communication for Multi-agent Pathfinding
计算机科学, 2025, 52(8): 317-325. <https://doi.org/10.11896/jsjcx.240900012>

融合机器学习预测和水波优化算法求解银行在线客服调度问题

卢雪琴¹ 谢歙铖² 唐燕³ 陈世昆³ 刘仰光³

1 浙江工业大学计算机科学与技术学院 杭州 310023

2 杭州师范大学信息科学与技术学院 杭州 311121

3 宁波财经学院金融与信息学院 浙江 宁波 315175

(luxueqin@nbufe.edu.cn)

摘要 在线客服的调度是银行客服中心的核心运营环节。高效的客服调度通过合理的人员配置与排班,确保客户能够获得服务,从而提升客户体验。然而,客户请求到达的随机性以及客服技能水平差异,使得在线客服调度问题变得复杂。对此,综合考虑客服技能等级、客户类型多样性以及匹配需求等因素,构建了一个以客户等待时间和运营成本最小化为优化目标的混合整数线性规划模型。针对客户需求的不确定性可能会导致客户需求和客服匹配困难,以及该问题在高维解空间中的求解复杂性,提出了一种融合机器学习预测与水波优化算法的混合方法来求解该客服调度问题。在该方法中,采用长短期记忆神经网络对客户到达量进行预测,充分捕捉其时间序列依赖性及其外部因素的影响。对于客服调度的混合整数规划模型,则通过一种结合强化学习 Q-learning 的水波优化算法进行高效求解。以浙江泰隆银行宁波分行在线客服中心的真实数据为基础进行实验,结果表明,所提方法在运营成本控制方面显著优于对比方法。进一步的灵敏度分析揭示:当预测准确率低于 90% 时,因客户到达量的不确定性,调度成本与客户等待时长均显著上升;而当预测准确率达到或超过 90% 后,系统性能的提升趋于平缓。这些发现不仅验证了高精度预测对调度效果的显著影响,还为实际应用中平衡预测模型复杂度与调度效率提供了理论基础和实践指导。

关键词: 在线客服调度; LSTM 神经网络; 强化学习; 水波优化算法

中图分类号 TP181

Integration of Machine Learning Prediction and Water Wave Optimization for Online Customer Service Representatives Scheduling in Bank Contact Centers

LU Xueqin¹, XIE Xicheng², TANG Yan³, CHEN Shikun³ and LIU Yangguang³

1 College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China

2 School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China

3 College of Finance & Information, Ningbo University of Finance & Economics, Ningbo, Zhejiang 315175, China

Abstract Online customer service representative scheduling is a crucial component of operational management in a bank's contact center. Optimized staffing and shift scheduling of customer service representatives ensure prompt customer responses, which significantly improves service efficiency and customer satisfaction. However, factors such as the uncertainty of customer request arrivals and variations in customer service skill levels make the online service representatives scheduling problem complex. Considering the practical challenges such as customer service representative skill levels, diversity of customer types, and matching requirements, this study proposes a mixed-integer linear programming model with the optimization objectives of minimizing customer waiting time and operational costs, and also presents a hybrid machine learning and water wave optimization(WWO) method to effectively solve the online customer service scheduling problem. In this method, a forecasting model based on long short-term memory neural networks is employed to predict customer arrival volumes, and this model can capture both time series dependencies and the influence of external factors. For the mixed-integer programming model of representative scheduling, WWO combining reinforcement learning Q-learning is used for efficient solution. This method leverages Q-learning to adaptively optimize neighborhood selection, enhancing the efficiency and quality of solutions. Based on real data from the contact center of Zhejiang

到稿日期:2025-05-21 返修日期:2025-07-20

基金项目:浙江省基础公益研究计划项目(LGF22G020002);浙江省教育厅高等学校访问学者专业发展项目(FX2023059);宁波市自然科学基金(2023J060)

This work was supported by the Zhejiang Provincial Basic Public Welfare Research Program(LGF22G020002), Zhejiang Provincial Department of Education College Visiting Scholars Professional Development Program(FX2023059) and Ningbo Natural Science Foundation(2023J060).

通信作者:刘仰光(liuyanguang@nbufe.edu.cn)

Tailong Bank's Ningbo branch, the experimental results show that the proposed method significantly outperforms comparative methods in terms of operational cost control. Further, sensitivity analysis reveals that when forecast accuracy drops below 90%, customer arrival uncertainty markedly increases scheduling costs and customer waiting times. Conversely, when accuracy reaches or exceeds 90%, performance improvements stabilize. These findings highlight the critical role of high-precision forecasting in effective scheduling and provide theoretical insights and practical guidance for balancing forecasting model complexity with scheduling efficiency in real-world applications.

Keywords Online customer service representatives scheduling, LSTM neural network, Reinforcement learning, Water wave optimization

1 引言

银行在线客服调度是指客服中心根据客户需求,合理决策每日各时段所需的人工客服的数量和每个客服的工作时段,从而有效应对来自手机银行 App、网页以及社交媒体平台等多种数字渠道的客户请求。随着客户在处理复杂或个性化问题时越来越倾向于寻求人工客服支持,银行在线客服调度面临着更大的运营挑战。

实际运营中,客户服务需求具有显著的波动性,常受到节假日、账单周期、政策变动及市场推广等因素的影响。例如,月底结算日或节假日期间,账户查询、支付争议等业务请求量通常大幅增加;政策变动或新产品推广也可能引发特定业务的集中涌入。这种需求不确定性显著增加了客服调度的复杂度:如果人员配置不足,可能造成客户等待时间延长、服务满意度下降,甚至引发客户流失;而人员配置冗余则会导致人力资源浪费和运营成本上升。

此外,在线客服通常采用“一对多”的服务模式,即一位客服可同时处理多个客户请求^[1]。相比传统电话客服的“一对一”服务,这种模式能显著缩短客户等待时间并降低人力成本^[2]。然而,当客户需求增加、客服并发处理能力达到上限时,客服响应速度下降、回复间隔延长,反而可能拉长客户整体服务时长^[3]。这一问题在客服技能水平不同时尤为显著,技能水平较低的客服难以有效平衡多任务需求,进一步加剧了服务效率的波动,增加了客户放弃服务的可能^[4-5]。

因此,在客户需求不确定性与客服能力差异并存的情况下,科学地配置人力资源与合理分配服务任务,成为在线客服调度的关键问题。

为解决上述问题,本文综合考虑客户到达的不确定性、客户类型的多样性、客服技能等级以及需求匹配等关键因素,构建了一个以客户等待时间与运营成本最小化为目标的混合整数线性规划模型,用于优化客服配置及其工作时段决策。

针对该问题求解的复杂性,本文提出了融合机器学习预测与元启发式优化的混合方法。在预测阶段,引入长短期记忆神经网络(Long Short-Term Memory, LSTM)对客户到达量进行预测,有效捕捉时间序列的动态依赖性,并纳入节假日、促销等外部扰动因素对请求量的影响。在优化阶段,本文在经典水波优化算法(Water Wave Optimization, WWO)的基础上,设计了4种邻域算子,并结合Q-learning强化学习机制,提出了Q-learning增强水波优化算法(QWWO)。该算法通过强化学习动态调整邻域算子的选择策略,提升局部搜索能力与收敛质量,在保证计算效率的同时获得更优的调度方案。

为验证所提模型和算法的性能,本文基于银行真实的业务数据构建测试用例。实验结果表明,QWWO在收敛速度与解的质量方面均优于多种对比算法,表现出较好的求解能力与稳健性。进一步,本文探讨了预测精度对调度性能的影响,通过构造不同预测准确率下的客户到达场景,以评估总的运营成本。结果显示:当预测精度不足时,客户到达量的不确定性增加,调度成本与等待时间大幅上升;而当预测准确率超过90%后,调度成本下降趋于稳定。这一分析不仅验证了预测精度对调度性能的关键影响,也为实际应用中在模型复杂度与调度效率之间的权衡提供了理论依据与实践参考。

本文的主要贡献如下:

- 1)提出了基于机器学习的客户需求预测模型,提升了对客户需求不确定性的响应能力,为银行在线客服的调度提供了较优的解决方案。
- 2)设计融合强化学习的水波优化算法(QWWO),将Q-learning的动态决策机制与水波优化算法的全局搜索能力相结合,优化邻域搜索策略,在提升求解效率的同时显著增强解的质量,为解决复杂客服调度问题提供了一种高性能的智能优化方法。

2 相关工作

客服调度是人员调度研究中的重要分支,其核心流程通常包括两个阶段^[6]:人员配置阶段决定各时段所需的客服人员数量,人员排班阶段则是在满足多种业务与制度性约束的前提下制定具体的工作时间表。随着服务业的快速发展和客户需求的日益复杂化,客服调度问题引起了学术界的广泛关注。本文主要聚焦于基于混合整数规划的精确建模方法及其求解算法的相关研究。表1总结了具有代表性的相关研究在调度目标设定、需求预测方法以及求解算法选择等多个维度的差异与特点^[7-21]。

从表1中可以看到,在客户需求预测建模方面,一些研究为简化问题,直接假设客户需求已知^[9,12,14,16-17];也有研究采用泊松分布等概率模型刻画需求的不确定性特征^[7,10-11,19-20]。随着数据可获取性的提升,越来越多的研究开始基于实际数据进行建模与仿真^[13,15,18],提高了模型的实用性。此外,部分工作将预测与人员配置决策相结合,借助基于智能体的排队模型,通过智能体建模增强了预测在调度中的决策支持能力^[21-22]。近年来,针对客户预测的研究进一步丰富了建模方法和应用场景。Henry等^[23]对传统时间序列方法与深度学习技术进行了系统性对比研究,发现传统时间序列方法在数

据有限的长期预测中表现稳定,而深度学习方法在短期预测中具有明显优势。上述研究对本文选择 LSTM 神经网络作为客户到达预测的方法具有重要的理论支撑和方法指导作用。

客服调度问题本质上是一个典型的 NP 难组合优化问题。在求解算法上,研究的技术路线经历了从精确算法到启发式算法,再到智能优化算法的发展演变。切平面算法等精确方法^[7]虽然能够获得全局最优解,但计算复杂度高,难以应对大规模问题。局部搜索^[8]和基础启发式算法^[10-13]虽能提高求解效率,却容易陷入局部最优,解的质量难以保证。为克服上述问题,研究逐步转向智能优化算法,包括自适应遗传

算法^[14-15]、禁忌搜索^[16-17]、模拟退火算法^[18]以及粒子群算法^[19]等,在提高求解效率的同时显著改善了解的质量,特别适用于大规模复杂问题。然而,这些优化算法通常依赖参数经验和固定搜索策略,泛化能力较弱,适应性有限。

近年来,融合机器学习的元启发式算法逐渐成为智能优化领域的研究热点。这类数据驱动的优化方法可通过对历史数据的学习,辅助搜索过程实现策略的自适应调整,有效增强算法的鲁棒性与泛化能力^[23]。这一融合范式已在生产调度、服务系统优化等领域得到应用^[24-27],这为本文中客服调度问题提供了新的求解思路与技术路径。

表 1 不同客服调度模型的比较

Table 1 Comparison of representatives scheduling models

方法	调度目标	调度算法	客户量预测/模型	应用场景
Chan 等 ^[7]	平衡成本与服务质量	切平面算法	齐次泊松过程	客服人员配置
Robbins 等 ^[8]	最小化总成本确保服务水平	局部搜索	均值和方差生成	人员配置与排班调度
Türker 等 ^[9]	控制劳动力成本满足最低服务	CPLEX 优化求解	假设为已知	班次固定的人员排班
Álvarez 等 ^[10]	最小化员工过载/短缺	启发式算法	非齐次泊松分布	灵活班次的人员排班
Vasiliyev 等 ^[11]	最小化人员需求缺口	局部搜索+启发式算法	泊松分布	班次规则、员工排班
Wu 等 ^[12]	最小化未被满足的总需求	启发式算法	假设为已知	人员配置与排班
Pandey 等 ^[13]	最小化人员冗余与短缺成本	启发式算法	蒙特卡洛模拟	人员配置
Ma 等 ^[14]	最小化预测与实际需求的差距	自适应遗传算法	假设为已知	人员工作时段安排
Yan 等 ^[15]	平衡员工工作负载	精英策略的遗传算法	实际数据模拟	人员排班
Rekik 等 ^[16]	最小化员工人数	禁忌搜索	假设为已知	班次灵活的人员排班
Wang 等 ^[17]	最小化人员成本	k-Opt 策略的禁忌搜索	假设为已知	人员配置和排班
Nthimo 等 ^[18]	最小化人员成本	邻域搜索+模拟退火算法	实际数据仿真	人员配置
Dhibar 等 ^[19]	最大化社会福利	粒子群+声搜索算法	泊松过程	人员状态
Xu 等 ^[20]	最小化总的员工人数/成本	人工蜂群算法	泊松过程	人员排班
Niu 等 ^[21]	确保服务水平标准的满足	二分法和逐步法	智能体建模技术	预测与人员配置协同

3 问题描述及建模

银行在线客服的服务时间通常为每天 7:00 至 22:00,覆盖客户主要活跃时段。假设每天在线服务时长被划分为 T 个时段,每个时段的时长根据运营需求可设为 30 分钟或 1 小时。

每类在线业务的客户可划分为高优先级客户和普通客户。在时段 $t \in \{1, 2, \dots, T\}$,高优先级客户和普通客户到达量分别是 $n_{1,t}$ 和 $n_{2,t}$,咨询消息数量分别服从参数为 $1/m_1$ 和 $1/m_2$ 的指数分布,且每条消息的发送间隔相互独立,服从参数为 $1/\mu$ 的指数分布。

当客户请求人工客服时,若有空闲客服的当前服务人数未达到上限,则客户可立即获得响应,否则客户需进入对应的等待队列。在服务过程中,客服同时处理着多个客户请求,导致对单个客户的响应速度下降,客户的服务时长延长。假设两类客户每次发送消息后,可接受的回复等待时间分别服从均值为 $1/\Delta T_1$ 和 $1/\Delta T_2$ 的指数分布。在等待队列中,两类客户可接受的等待时间分别服从均值为 $1/\Delta T_1$ 和 $1/\Delta T_2$ 的指数分布。如果客户的等待时间超过阈值,客户将放弃服务。单位时间等待成本分别为 c_1 和 c_2 ,放弃服务将产生相应的惩罚成本,惩罚系数分别是 α_1 和 α_2 。

客服人员根据技能熟练程度分为高级、中级和初级 3 类。每天 3 类客服的工作时长均不超过 T^{\max} 个时段,包含在线服务和休息时间。其中,连续在线服务时段不超过 \hat{T} 个,总在线服务时段不少于 T^{\min} 个。3 类客服可用人数分别不超过

N_1, N_2 和 N_3 ,同时服务客户数的最大值分别是 L_1, L_2 和 L_3 ,回复客户消息的时间分别服从均值为 $1/\theta_1, 1/\theta_2$ 和 $1/\theta_3$ 的指数分布,人力成本分别为 s_1, s_2 和 s_3 。

为构建在线客服调度模型,本文一对多服务形式作如下假设。

1) 客户与客服的交互严格遵循“你来我往”的单条消息往返形式,客户在收到客服对上一条消息的回复后,才继续发送消息;客服每次只能回复一位客户。

2) 客服以“非空闲”方式进行服务,当完成某个客户的服务后,若队列中有客户等待,则立即服务队首客户。仅当所有的客服均处于忙碌状态且同时服务的人数达到上限时,新客户才会进入等待队列。

3) 客户从接受服务到结束,始终由同一个客服接待。每位客服仅服务分配给自己的客户。

为了便于后续的描述和数学建模,本文对模型中涉及的各项变量和参数进行统一符号定义,具体索引符号如表 2 所列,输入变量及其含义如表 3 所列。

客服调度旨在合理配置不同技能等级的客服人员及其工作时段,以应对需求波动,实现服务成本与服务质量的最优平衡。因此,本文构建的混合整数规划模型包含各类客服人员配置变量和每位客服工作时段的两进制变量两个决策变量。

假设人员配置变量为 $X = \{x_u, u \in \{1, 2, 3\}\}$,其中 x_u 表示 u 类客服的人数。工作时段变量为 $Y = \{y_{ukt}, u \in \{1, 2, 3\}, k \in \{1, 2, \dots, x_u\}, t \in \{1, 2, \dots, T\}\}$,其中 $y_{ukt} \in \{0, 1\}$ 表示 u 类中客服 k 在时段 t 是否提供在线服务,若提供服务,则 $y_{ukt} =$

1, 否则 $y_{ukt} = 0$ 。

为了表示客服的连续工作时段, 引入辅助变量 z_{ukt}, z_{ukt} 表示 u 类中客服 k 到时段 t 为止的连续工作时间总和。 z_{ukt} 根据客服在各时段的工作状态递推更新, 如果客服在时段 t 工作, 则 z_{ukt} 在上一时段的基础上加 1; 否则重置为 0。初始值为 $z_{uk1} = y_{uk1}$, 当 $t \geq 2$ 时, z_{ukt} 的更新公式为:

$$z_{ukt} = \begin{cases} z_{uk(t-1)} + y_{ukt}, & \text{if } y_{ukt} = 1 \\ 0, & \text{if } y_{ukt} = 0 \end{cases} \quad (1)$$

建立数学模型的过程中涉及到的中间变量如表 4 所列。

表 2 索引符号及意义

Table 2 Indexes and their meanings

参数符号	意义
u	客服等级的索引编号, $u = \{1, 2, 3\}$ 分别表示高级、中级和初级客服
i	客户类别的索引编号, $i = \{1, 2\}$, 分别表示高优先级客户和普通客户
t	时间段的索引编号, $t = \{1, 2, \dots, T\}$
j	客户的索引编号
k	客服的索引编号

表 3 输入变量符号及意义

Table 3 Input variables and their meanings

参数符号	意义
$n_{i,t}$	i 类客户在时段 t 内的到达数量
M_{j_i}	客户 j_i 咨询消息的数量, 服从均值为 $1/m_{j_i}$ 的指数分布
$d_{j_i,m}$	客户 j_i 在服务过程中发送第 m 消息的时间, 服从均值为 $1/\mu$ 的指数分布, $m \in \{1, 2, \dots, M_{j_i}\}$
ΔT_{j_i}	等待队列中, 客户 j_i 愿意等待的时间, 服从均值为 $1/\Delta T_{j_i}$ 的指数分布
$\hat{\Delta T}_{j_i}$	服务过程中, 客户 j_i 发送一条消息后愿意等待的时间, 服从均值为 $1/\hat{\Delta T}_{j_i}$ 指数分布
c_i	i 类客户单位时间的等待成本
α_i	i 类客户放弃服务产生的惩罚系数
T	每个工作日时间段的数量
T^{\max}	客服的最大工作时段数
\hat{T}	客服连续服务的最大时段数
T^{\min}	客服的最小服务时段数
L_u	u 类客服可同时服务的最大客户数
N_u	u 类客服的人数上限
r_{k_u}	客服 k_u 回复每条消息的时间, 服从均值为 $1/\theta_u$ 的指数分布
s_u	u 类客服的人力成本, 以单位时间内的工资计

表 4 中间变量符号及意义

Table 4 Intermediate variables and their meanings

参数符号	意义
$t_{j_i}^a$	客户 j_i 的到达时间
$t_{j_i}^s$	客户 j_i 开始接受服务的时间
$t_{j_i,m}^d$	客户 j_i 发送第 m 条消息的时间
$t_{j_i,m}^{w_1}$	客户 j_i 发送第 m 条消息后的等待时间
$t_{j_i}^c$	客户 j_i 完成或者放弃服务的时间
$t_{j_i}^{w_2}(q)$	客户 j_i 在队列中的等待时间
$t_{j_i}^{w_2}(s)$	客户 j_i 在服务中的等待时间
$l_{k_u}(\tau)$	客服 k_u 在 τ 时刻同时服务的客户数
$t_{k_u,j_i,m}^s$	客服 k_u 开始回复客户 j_i 第 m 条消息的时间
$t_{k_u,j_i,m}^f$	客服 k_u 完成回复客户 j_i 第 m 条消息的时间
$t_{k_u}(n, n-1)$	客服 k_u 服务的客户数由 n 变为 $n-1$ 的时间
q_i	放弃服务的 i 类客户总数

在线客服模型以减少客户等待时间和客户流失, 同时控制人力成本为优化目标。因此, 目标函数包含 3 部分成本: 客户等待成本 C_w 、客户放弃服务的惩罚成本 C_a , 以及客服的

人力成本 C_s 。

假设在时段 t 内, 客户的到达近似为齐次泊松过程, 则客户 j_i 的到达时间为:

$$t_{j_i}^a = (t-1)\Delta + \frac{-\ln(U)}{n_{i,t}} \quad (2)$$

其中, Δ 指每个时段的时长, U 是一个均匀分布的随机数 $U \in [0, 1]$ 。

时段 t 内, 提供在线服务的各类客服数量 $N_{u,t}$ 可表示为:

$$N_{u,t} = \sum_{k=1}^{x_u} y_{ukt} \quad (3)$$

客户到达后, 按照“先到先服务”的原则, 将高优先级客户分配给高级客服, 普通客户分配给中级或初级客服。为了提高客服的利用效率, 减少高优先级客户的等待时间, 本文借鉴文献[28]中的相关研究, 设计了一个基于等待时间调整服务优先级的客户分配策略, 允许高优先级客户在一定条件下进入中级客服的服务范围, 具体规则如下:

1) 当有空闲的高级客服时, 直接分配。

2) 当所有高级客服都处于忙碌状态时, 且普通客户的等待队列长度 $Q_{e,t}$ 小于阈值 $Q_{th,t}$, 则分配给空闲的中级客服服务, 否则进入高优先级等待队列。

客户分配过程如图 1 所示。

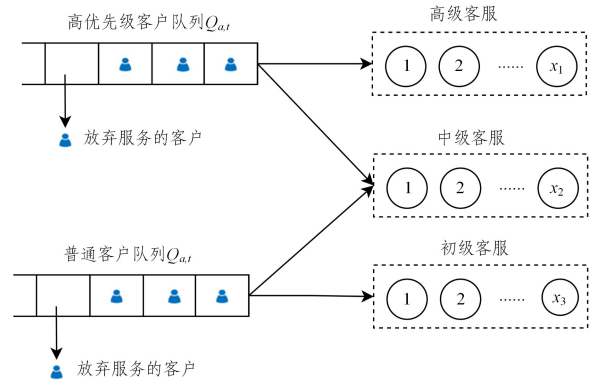


图 1 客户分配过程示意图

Fig. 1 Workflow of customer assignment in a contact center

阈值 $Q_{th,t}$ 的更新机制如下:

$$Q_{th,t} = \left\lceil \rho_t \frac{\sum_{u=2}^3 N_{u,t} \times L_u \times \theta_u}{\sum_{u=2}^3 N_{u,t}} \right\rceil \quad (4)$$

其中, $\lceil \cdot \rceil$ 表示向上取整, ρ_t 表示高级客服全部繁忙的概率。

$$\rho_t = P \left\{ \frac{n_{1,t}}{N_{1,t} \times L_1 \times \theta_1} \geq 1 \right\} \quad (5)$$

在第一个时段, 初始阈值为:

$$Q_{th,1} = \left\lceil \frac{\sum_{u=2}^3 N_{u,1} \times L_u}{2} \right\rceil \quad (6)$$

在每个时段 t 开始前计算阈值 $Q_{th,t}$ 。当高优先级客户 j_i 在 $t_{j_i}^a$ 到达时, 如果有空闲的高级客服, 则分配给空闲的高级客服 k_1 :

$$k_1 = \arg \min_{k \in \{1, 2, \dots, N_{1,t}\}} \{l_{k_1}(t_{j_i}^a), l_{k_1}(t_{j_i}^a) < L_1\} \quad (7)$$

如果高级客服都处于忙碌状态, 并且普通客户等待队列的长度没有超过阈值, 则溢出给空闲的中级客服 k_2 :

$$k_2 = \arg \min_{k \in \{1,2,\dots,N_{2,t}\}} \{l_{k_2}(t_{j_1}^a), l_{k_2}(t_{j_1}^a) < L_2 \text{ 且 } Q_{r,t} < Q_{th,t}\} \quad (8)$$

如果普通客户等待队列的长度超过阈值,则客户 j_1 进入高优先级等待队列。

当普通客户 j_2 在 $t_{j_2}^a$ 时刻到达时,如果有空闲的中级或者初级客服,则分配给空闲的客服 k_u :

$$k_u = \arg \min_{k \in \{1,2,\dots,N_{u,t}\}} \{l_{k_u}(t_{j_2}^a), l_{k_u}(t_{j_2}^a) < L_u, u \in \{2,3\}\} \quad (9)$$

否则,客户 j_2 进入普通等待队列。

当客户 j_i 被分配给客服 k_u 进行服务时,其服务人数 $l_{k_u}(t_{j_i}^a)$ 增加 1。客户开始接受服务的时间为 $t_{j_i}^s = t_{j_i}^a$, 在队列中的等待时间为 $t_{j_i}^w(q) = 0$ 。

当客户 j_i 进入等待队列时,需等待直至有客服完成当前客户的服务,释放出空闲服务能力。若有多位客服都可服务,则优先分配给最早完成服务的客服。该分配机制在高优先级队列和普通客户队列中一致。

此时,队列中客户 j_i 被分配的客服 k_u 可以表示为:

$$k_u = \arg \min_{k \in \{k: l_{k_u} < L_u\}} \{t_{k_u}(L_u, L_u - 1)\} \quad (10)$$

客户 j_i 开始接受客服 k_u 服务的时间可以表示为:

$$t_{j_i}^s = \max\{t_{j_i}^a, t_{k_u}(L_u, L_u - 1)\} \quad (11)$$

客户 j_i 在队列中的等待时间为:

$$t_{j_i}^w(q) = t_{j_i}^s - t_{j_i}^a \quad (12)$$

当客户 j_i 的等待时间超过其愿意等待的时间 ΔT_{j_i} , 客户放弃服务。因此,等待队列中离开客户的总数量为:

$$q_i + = \begin{cases} 1, & \text{if } t_{j_i}^w(q) \geq \Delta T_{j_i} \\ 0, & \text{else} \end{cases} \quad (13)$$

当客户 j_i 开始与客服 k_u 进行消息交互时,客服 k_u 的服务人数 $l_{k_u}(t_{j_i}^s)$ 增加 1。在服务过程中,客服与客户采用“一来一往”的消息交互方式。假设客服 k_u 当前服务 l 个客户,为了确保交互过程的有序性,客服根据消息到达时间的先后顺序依次处理各个客户的请求,即优先回复最早发送消息的客户。客服回复消息的过程如图 2 所示。

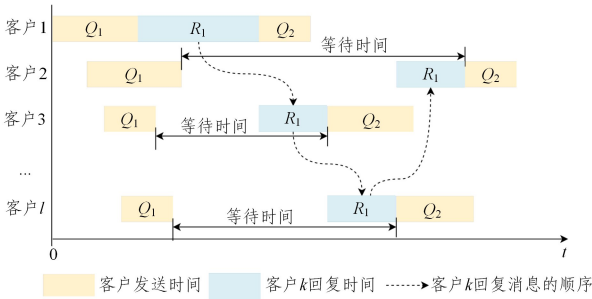


图 2 客服回复消息过程

Fig. 2 Message response process by service representatives

假设客服 k_u 在 τ 同时服务的活跃客户队列为 $Q_{k_u}(\tau)$ 。初始状态客户队列为空,即 $Q_{k_u}(\tau) = \emptyset$ 。当客户 j_i 分配给客服 k_u , $Q_{k_u}(t_{j_i}^s) = Q_{k_u}(\tau) \cup j_i$, 客户 j_i 成为队列中的第 l 位客户:

$$l = |Q_{k_u}(t_{j_i}^s)| + 1 \quad (14)$$

客户 j_i 第 1 条消息的发送结束时间为:

$$t_{j_i,1}^d = t_{j_i}^s + d_{j_i,1} \quad (15)$$

所有活跃客户的待回复消息按发送结束时间排序,形成一个消息队列 $Q_{msg}(\tau)$, 客服 k_u 依次按照消息到达的时间顺序进行回复。

对于客户 j_i 的第 1 条消息,如果客户 j_i 是队列中的第一个客户,即 $l=1$,则在客户发送完消息后,客服立刻回复。

$$t_{k_u, j_i, 1}^s = t_{j_i, 1}^d \quad (16)$$

如果不是第一个客户,即 $l > 1$,则客服的回复时间取决于回复完前一条消息的时间。

$$t_{k_u, j_i, 1}^s = \max\{t_{j_i, 1}^d, t_{k_u, l-1, 1}^f\} \quad (17)$$

客服 k_u 完成客户 j_i 的第 1 条消息回复的时间:

$$t_{k_u, j_i, 1}^f = t_{k_u, j_i, 1}^s + r_{k_u, 1} \quad (18)$$

客户 j_i 的第 m 条消息要在客服完成第 $m-1$ 条消息的回复后才能发送,客户在收到回复后立即发送下一条消息。因此第 m 条消息的发送结束时间为:

$$t_{j_i, m}^d = t_{k_u, j_i, m-1}^f + d_{j_i, m} \quad (19)$$

客服 k_u 开始回复客户 j_i 的第 m 条消息的时间:

$$t_{k_u, j_i, m}^s = \max\{t_{j_i, m}^d, t_{k_u, m-1}^f\} \quad (20)$$

其中, $t_{k_u, m}^f$ 表示消息队列 $Q_{msg}(\tau)$ 中客户 j_i 第 m 条消息前一条消息的完成时间。

客服 k_u 完成客户 j_i 的第 m 条消息回复的时间:

$$t_{k_u, j_i, m}^f = t_{k_u, j_i, m}^s + r_{k_u, m} \quad (21)$$

客户 j_i 在第 m 条消息发送后的等待时间:

$$t_{j_i, m}^w = t_{k_u, j_i, m}^f - t_{j_i, m}^d \quad (22)$$

当客户 j_i 的等待时间超过阈值 ΔT_{j_i} 时,客户放弃服务:

$$q_i + = \begin{cases} 1, & \text{if } t_{j_i, m}^w \geq \Delta T_{j_i} \\ 0, & \text{else} \end{cases} \quad (23)$$

记客户放弃服务时发送的消息是第 m^* 条,则客户在服务过程中的等待时间:

$$t_{j_i}^w(s) = \begin{cases} \sum_{m=1}^{M_{j_i}} t_{j_i, m}^w, & \text{if } t_{j_i, m}^w \leq \Delta T_{j_i} \\ \sum_{m=1}^{m^*-1} t_{j_i, m}^w + \Delta T_{j_i}, & \text{else} \end{cases} \quad (24)$$

客服 k_u 完成客户 j_i 的全部消息回复的时间:

$$t_{k_u, j_i}^f = t_{k_u, j_i, m^*}^f + r_{k_u, M_{j_i}} \quad (25)$$

在所有时段内,客户放弃导致的惩罚成本 C_a 可表示为:

$$C_a = \sum_{t=1}^T \sum_{i=1}^2 \alpha_i q_i \quad (26)$$

客户的等待成本 C_w 可表示为:

$$C_w = \sum_{t=1}^T \sum_{i=1}^2 \sum_{j=1}^{n_{i,t}} c_i (\min\{t_{j_i}^w(q), \Delta T_{j_i}\} + t_{j_i}^w(s)) \quad (27)$$

若客户 j_i 每次发送完消息后的等待时间都未超过阈值,则客户将完成所有的消息交互,并在服务完成后离开,客服完成对客户服务的最后一条消息回复的时间即为客户离开服务的时间。因此,客户离开的时间:

$$t_{j_i}^c = \begin{cases} t_{k_u, j_i, M_{j_i}}^f, & \text{if } t_{j_i, m}^w \leq \Delta T_{j_i} \\ t_{k_u, j_i, m^*-1}^f + \Delta T_{j_i}, & \text{else} \end{cases} \quad (28)$$

一旦客户 j_i 离开,客服 k_u 当前服务人数减 1,则有:

$$l_{k_u}(t_{j_i}^c) - = 1 \quad (29)$$

$$t_{k_u}(l, l-1) = t_{j_i}^c \quad (30)$$

每天客服的人力成本 C_s 可表示为:

$$C_s = \sum_{t=1}^T \sum_{u=1}^3 \sum_{k=1}^{N_{uk}t} s_u y_{ukt} \quad (31)$$

最终, 总成本为:

$$f = C_w + C_a + C_s \quad (32)$$

因此, 客服调度模型的目标函数可表示为:

$$\min f(X, Y) = C_s(X, Y) + C_a(X, Y) + C_w(X, Y) \quad (33)$$

$$\text{s. t. } 1 \leq x_u \leq N_u, \forall u \quad (34)$$

$$\sum_{u=1}^3 \sum_{k=1}^{x_u} y_{ukt} \leq 1, \forall u, k, t \quad (35)$$

$$T^{\min} \leq \sum_{t=1}^T \leq T^{\max}, \forall u, k, t \quad (36)$$

$$z_{ukt} \leq \hat{T}, \forall u, k, t > 2 \quad (37)$$

$$x_u \in \mathbb{Z}^+, y_{ukt} \in \{0, 1\} \quad (38)$$

其中, 式(34)表示配置的各类客服的数量不超过可用的总的客服数, 式(35)约束每个客服在每个时段分配一次, 式(36)表示客服的总服务时长的约束, 式(37)约束每个客服的连续服务时长不超过 \hat{T} , 式(38)表示每个参数的取值范围, \mathbb{Z}^+ 表示非负整数。

对于目标函数式(27), 客服的人力成本 $C_s(X, Y)$ 的计算涉及所有客服代表在各时段的工作状态评估, 其复杂度为 $O(\sum_{u=1}^3 N_u \times T)$, 放弃服务的惩罚成本 $C_a(X, Y)$ 和客户等待成本 $C_w(X, Y)$ 均需要遍历两类客户类型在各时段的服务状况, 复杂度均为 $O(T)$, 因此目标函数的总体计算复杂度为:

$$O(f) = O(\sum_{u=1}^3 N_u \times T + T + T) = O(\sum_{u=1}^3 N_u \times T) \quad (39)$$

4 混合机器学习和启发式优化的方法

首先, 基于历史数据挖掘和建模, 利用机器学习预测调度模型中不确定性变量的分布特征。在此基础上, 结合混合 Q-learning 的水波优化算法, 对客服人员的配置及其工作时段进行联合优化, 从而实现高效调度。

4.1 相关参数预测

4.1.1 基于 LSTM 的客户量预测

银行客户的到达模式通常具有明显的时间依赖性和周期性特征。这种规律性体现在多个时间维度上, 包括日内的高峰与低谷、每日需求的波动、工作日与周末的差异, 以及季节性变化。例如, 在工作日的早晚高峰、午休时间及下班后, 客户咨询需求往往显著增加, 而周末或节假日的咨询量则明显减少^[29]。季节性因素同样会对客户到达产生重要影响, 如在年底假期期间, 贷款与理财类咨询明显增加, 而年初则常见于账户信息更新的集中办理。

此外, 客户到达过程也具有一定的随机性和波动性^[30], 主要受促销活动、金融市场波动、政策调整等外部事件影响。这些因素往往在短时间内导致客户到达量显著偏离常规模式。例如, 购物节期间与在线支付相关的咨询量激增, 而节假日休市则会使部分业务咨询量骤减。因此, 准确的客户量预测模型需充分纳入这些外部变量特征。

基于对历史数据的分析与文献综述, 本文归纳了影响客

户到达量的主要特征变量。

1) 时间特征, 包括月份、周数、日期、星期、昼夜、时间间隔编号、是否为法定假日及其前后日期。

2) 历史到达数据, 如前一日客户量、七日前客户量, 以及过去 30 天的移动平均值。

3) 外部影响因素, 包括节假日、促销活动、政策调整等。

面对如此复杂且多样化的影响因素, 客户到达量预测模型不仅需捕捉周期性变化, 还需具备识别外部冲击的能力。为提升预测精度, 本文尝试了多种方法, 包括神经网络^[31]、回声状态网络^[32]、泊松线性回归模型^[33]、梯度提升决策树^[34]和随机森林等模型^[29], 通过对比各模型的预测性能, 最终选择利用长短时记忆网络模型(LSTM)作为客户到达量的预测工具。LSTM 模型在处理时间序列数据、建模长期依赖关系方面表现优异, 能够有效捕捉客户到达模式中的动态性与复杂性, 适用于在线客服中的客户流量预测任务。

LSTM 基本结构框架如图 3 所示。其核心架构依托于门控机制来控制信息的传输与留存, 主要包含遗忘门、输入门、候选记忆单元、记忆单元状态以及输出门这 5 个关键部分。图中 x_t 表示 t 时刻的输入向量, h_{t-1} 是上一个时刻的隐藏状态, 即 LSTM 的输出。

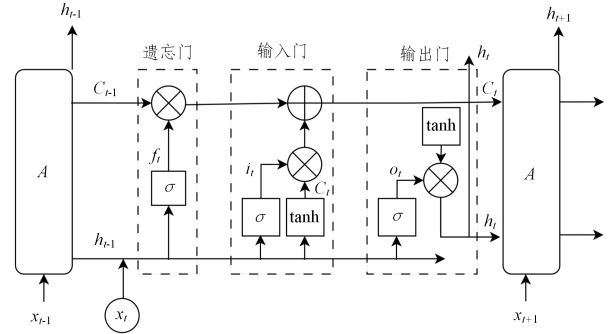


图 3 LSTM 基本结构框架图^[35]

Fig. 3 Schematic diagram of LSTM structure^[35]

遗忘门 f_t 控制保留多少先前的记忆状态, 其计算式为:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t]) + b_f \quad (40)$$

输入门 i_t 决定当前输入信息对记忆单元的贡献程度, 表达式为:

$$f_t = \sigma(W_i \cdot [h_{t-1}, x_t]) + b_i \quad (41)$$

候选记忆单元 \tilde{C}_t 生成当前的新信息候选值, 表达式为:

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t]) + b_c \quad (42)$$

其中, \tanh 是双曲正切函数, 将输入映射到 $(-1, 1)$ 。将 i_t 和 \tilde{C}_t 相乘, 随后加上遗忘门信息以获取新的记忆单元 C_t 。

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (43)$$

输出门 o_t 的结果是由最终 C_t 和上一时刻的输出 h_{t-1} 以及当前时刻的输入向量 x_t 确定, 其表达式为:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t]) + b_o \quad (44)$$

式(40)–式(44)中, $\sigma(\cdot)$ 为 sigmoid 激活函数; W_f, W_i, W_c, W_o 是权重矩阵; b_f, b_i, b_c, b_o 是偏置项。随着训练数据的不断输入, 这些权重和偏置项通过反向传播算法进行逐步优化, 直到模型收敛位置。

LSTM 的最终输出即为当前时刻的隐藏状态 h_t :

$$h_t = o_t \cdot \tanh(C_t) \quad (45)$$

4.1.2 其他参数预测

各类客户咨询消息的平均数量 m_i 、发送每条消息的平均时间 μ 及愿意等待的平均时间 ΔT_i 和 $\hat{\Delta T}_i$, 以及各类客服回复每条消息的平均时间 θ_i 等变量, 均具有一定的不确定性。为提高客服调度模型的精度, 本文通过对历史数据的挖掘与分析, 采用传统的数据处理方法(如分组、聚合与分层分析)对变量进行估计。

首先, 基于历史交互记录提取客户身份、问题类别、客户与客服的交互时间戳、客户是否中途放弃服务、离开系统的时间点等特征信息; 进一步提取每条消息的发送者、消息时间戳和客户类别。通过识别连续两条消息之间的发送者与时间间隔, 计算客服回复消息的平均响应时间。

其次, 根据客服级别对数据进行分组, 统计各类客服的消息回复时长, 并计算其平均回复时间。同时, 根据时间维度对数据进行分层分析, 识别各级客服在不同时段内响应时间的变化趋势, 以捕捉其中的周期性或趋势性特征。

这些不确定变量的动态变化直接影响到客服配置和排班策略。通过对其在历史数据中的估计和建模, 可显著提升调度模型在实际应用中的适应性与准确性, 从而实现更高效的资源调度与服务保障。

4.2 混合强化学习的水波优化算法设计

本文提出了一种融合强化学习的水波优化启发式算法(QWWO), 以水波优化算法作为主要搜索框架, 通过集成邻域搜索策略提升局部解的挖掘能力, 同时引入强化学习(Q-learning)进一步增强局部搜索的适应性, 从而提高求解精度。

4.2.1 算法整体框架

算法首先根据问题特征生成初始解种群, 随后经过传播、折射与碎浪等算子, 结合邻域搜索与强化学习机制对

种群不断演化。为保障算法的收敛性与可操作性, 本文将最大迭代次数设为终止条件。算法流程如图 4 所示。流程图中解的适应度值 $f(X, Y)$ 根据目标函数式(33)进行计算。

QWWO 算法的伪代码如算法 1 所示。在传播操作中, 算子要生成各类客服的数量和工作时段, 因此传播操作的时间复杂度是 $O(n \cdot d) + O(f)$, 其中 d 是解的维度数。4 个邻域搜索算子的时间复杂度分别为 $O(n^2)$, $O(n)$, $O(n^2)$ 和 $O(n)$ 。基于 Q-learning 的碎浪操作中, 状态空间 S 和动作空间 A 均为有限小规模集合, 因此每次碎浪操作的复杂度为 $O(1)$, 整体复杂度 $O(n)$ 。假设种群规模是 NP , 迭代次数为 G , 则算法的整体复杂度为 $G \cdot NP[O(n^2) + O(f)]$ 。

算法 1 QWWO 算法

输入: $NP, \alpha, \lambda_{\min}, \lambda_{\max}, h_{\max}$

输出: 最优解 (X^*, Y^*)

1. 初始化种群, 根据算法 2, 生成 NP 个初始解
2. 根据目标函数计算种群中每个解的适应度值, 并找到当前最优解 (X^*, Y^*)
3. while 终止条件不满足 do
4. for 种群中的每个解 (X, Y) do
5. 根据算法 3 生成新的解 (X', Y') /* 传播操作 */
6. if $f(X', Y') < f(X, Y)$ then
7. 根据算法 4 生成新解 (X'', Y'') /* 碎浪操作 */
8. if $f(X'', Y'') < f(X^*, Y^*)$ then
9. $(X^*, Y^*) \leftarrow (X'', Y'')$ /* 当前最优解 */
10. else
11. $h_{(X, Y)} \leftarrow h_{(X, Y)} - 1$ /* 波高减少 1 */
12. if $h_{(X, Y)} = 0$ then
13. 根据算法 5 生成新解 /* 折射操作 */
14. 更新种群中所有解的波长
15. return 当前最优解 (X^*, Y^*)

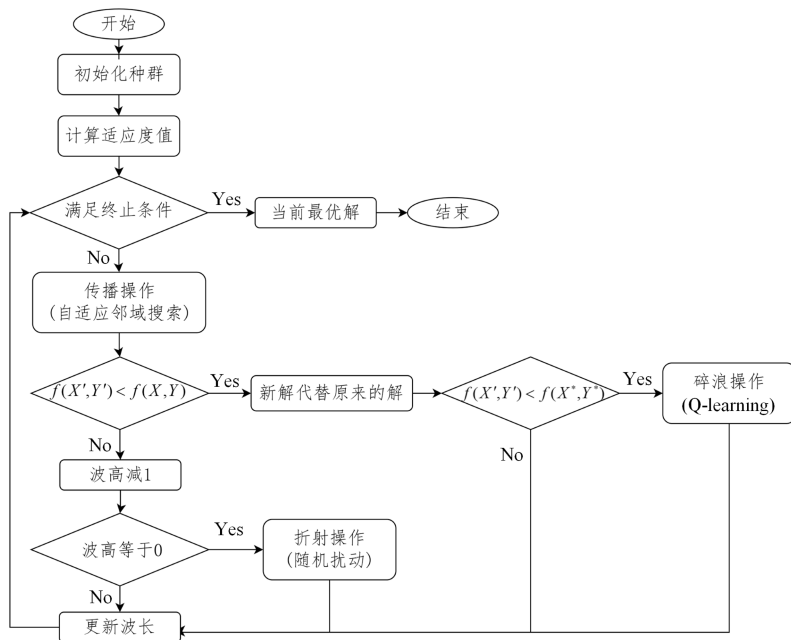


图 4 QWWO 算法流程图

Fig. 4 Flowchart of QWWO algorithm

4.2.2 解的编码与初始化

客服调度问题涉及多类客服和多个工作时段,构成一个高维离散的优化空间。其中,各类客服的数量通过一个整数向量 $\mathbf{X} = \{x_u, u \in \{1, 2, 3\}\}$ 表示, x_u 表示每天配置的 u 类客服的人数。采用一个二进制向量 \mathbf{y}_{ukt} 表示每个客服在各个时段的工作状态。图 5 展示了客服工作时段的状态编码示意。

所有客服的排班方案可以用一个 $N \times T$ 的矩阵表示,其中 $N = \sum_{u=1}^3 x_u$ 表示各类客服的总数, T 表示每天提供在线服务的总工作时段数量。矩阵中元素的取值为 $\{-1, 0, 1\}$, 其中, 1 表示客服在该时段在服务状态, 0 表示客服在休息状态, -1 表示该客服在非工作状态(即不上班)。

为确保初始种群中的解能满足基本可行性,同时为后续优化提供多样化的起点,本文充分考虑决策变量的特征和客服工作时段的时间约束,设计了一种启发式算法用于随机生成初始解。

初始化过程主要分成两个阶段:首先随机确定各类客服数量,其次为每个客服生成符合时间约束的工作时段。具体流程如下。

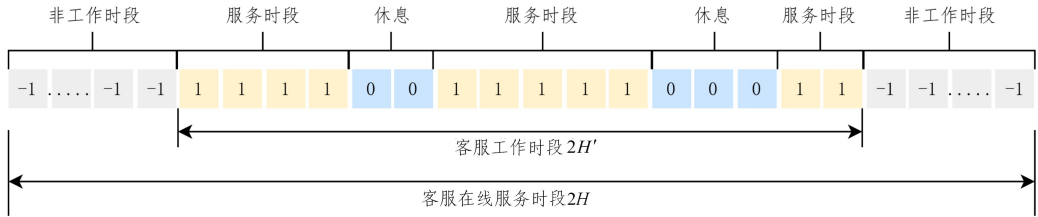


图 5 客服工作时段状态

Fig. 5 Example shift for a customer service

算法 2 初始化种群的算法

输入: $T, T^{\min}, T^{\max}, \mu, N_u, N_u^{\min}$

输出: 初始种群

```

1. for n=1 to NP do:
2.   for u=1 to 3 do:
3.      $x_u = \text{rand int}(N_u^{\min}, N_u) / *$  随机客服数  $*/$ 
4.   初始化矩阵  $\mathbf{Y}$ , 所有元素为 -1
5.   for k=1 to  $\sum_{u=1}^3 x_u$  do:
6.     剩余工作时长  $T_r = T^{\max}$ 
7.     工作起始时段  $t_s = \text{rand int}(1, T - T^{\max} + 1)$ 
8.     当前时段  $t = t_s$ 
9.     while True do:
10.      if  $T_r > 0$  且  $t < T$ :
11.         $D = \text{rand int}(1, \min(\hat{T}, T_r))$ 
12.        if  $t + D - 1 < T$ :
13.          for  $s = t$  to  $t + D - 1$  do:
14.            设置服务时段  $Y[u][k][s] = 1$ 
15.            更新剩余工作时长  $T_r = T_r - D$ 
16.             $t = t + D$ 
17.          if  $T_r > 0$ :  $/*$  随机生成一段休息时长  $*/$ 
18.             $R = \text{randint}(1, \min(T - T_r, T - t + 1))$ 
19.            for  $s = t$  to  $t + R - 1$  do:
20.              设置休息时段  $Y[u][k][s] = 0$ 
21.              更新剩余工作时长  $T_r = T_r - R$ 
22.             $t = t + R$ 

```

Step1(初始化客服数量) 对于每一类客服,生成 $[N_u^{\min}, N_u]$ 内的随机整数 x_u , 其中 N_u^{\min} 表示历史数据中这一类客服的最小值。由此确定各类客服的总人数 N 。

Step2(初始化工作时段矩阵) 初始时,矩阵中所有元素 $y_{ukt} = -1$, 表示未分配工作时段。随机生成服务时长 $D = \text{rand int}(1, \hat{T})$ 和工作起始时段 $t_s = \text{rand int}(1, T - T^{\max} + 1)$, 并设置当前时段为起始工作时段 $t = t_s$ 。

Step3(分配连续服务时长) 对于每个客服,当剩余总的工作时长和服务时长大于 0,且当前时段未到最后时段,即 $t < T$,随机生成一段符合约束的连续服务时长 D ,如果 D 小于剩余的服务时长,并且 $t + D - 1 < T$,则将这一时段内客服的状态设置为服务状态,即 $y_{uks} = 1$, 其中 $s \in [t, t + D - 1]$ 。当剩余总的工作时长大于 0,并且当前时段未到最后时段,则插入随机的休息时段。

Step4 重复以上操作,直到总服务时长小于 0。

Step5 如果总的剩余工作时长仍大于 0,则设置剩余的工作时段 $y_{ukt} = 0$ 。未分配的时段保持 $y_{ukt} = -1$ 。

Step6 重复以上步骤,执行 NP 次循环生成初始种群。

```

23.   else break  $/*$  跳出当前循环  $*/$ 
24.   if  $T_r > 0$ :  $/*$  仍有未分配的时段  $*/$ 
25.     for  $s = t$  to  $t + T_r - 1$  do:
26.       设置为休息时段  $Y[u][k][s] = 0$ 
27.   return 初始化种群

```

算法 2 从随机起始点开始,交替分配服务和休息时段,确保生成的初始解能满足时段约束;通过随机选择起始时段、服务时长和休息时长,保证解的多样性。

4.2.3 邻域结构

为了进一步优化解中客服工作时段安排,本文根据文献[36]的研究,结合排班问题常用的邻域结构,设计了一组针对客服工作时段调整的邻域算子。这些邻域操作既涵盖了客服之间的排班交换,又包括了单个客服自身的排班调整,有助于在局部范围内高效探索解空间,提升算法的搜索质量。具体描述如下:

NS1(整体交换) 随机选取两个不同类别的客服,对其工作时段序列进行整体互换,直到解的适应度值提升或者交换次数达到上限 $n^2/4$ 。

NS2(时段平移) 随机选取某个客服,将其工作时段序列整体前移或者后移一个时段,直到解的适应度值提升或者交换次数达到上限 n 。

NS3(随机重排) 随机选择某个客服,从其排班序列中随机挑选若干工作时段,然后将它们插入序列中其他随机选择的位置,直到解的适应度值提升或者交换次数达到

上限 $n^2/4$ 。

NS4(时段交换) 随机选取某个客服的一个服务时段(值为1)和一个休息时段(值为0),交换这两个时段的状态,直到解的适应度值提升或者交换次数达到上限 n 。

在邻域算子 NS3 和 NS4 中,若生成的解违反了连续服务最长时间的约束条件,需引入修正机制对解进行调整。具体做法是:在违反约束的位置,将服务时段转换为休息时段,即将对应排班矩阵中的值从 1 修改为 0,以恢复对约束的满足。该修正方式在确保可行性的同时,保留了原始邻域操作的优化趋势。

4.2.4 自适应邻域搜索策略的传播操作

在 WWO 算法中,种群中的每个解被类比为具有波长 λ 和波高 h 的水波。传播操作的核心思想是为每个解指定与其适应度相关的波长,并使得每个解的搜索范围与其波长成正比。具体而言,越优的解,对应的波长越短,波高越高,搜索范围也相应越窄。

客服排班问题是一个最小化目标函数问题,根据文献[37]定义解的波长与适应度(目标函数)的关系为:

$$\lambda(X,Y) = \lambda_{\max} - (\lambda_{\max} - \lambda_{\min}) \times \alpha \frac{f(X,Y) - f_{\min} + \epsilon}{f_{\max} - f_{\min} + \epsilon} \quad (46)$$

其中,当前解的目标函数值是 $f(X,Y)$;当前种群中最优解的目标函数值为 f_{\min} ,最差解的目标值为 f_{\max} ; α 是波长缩减系数,取值为 1.0026; $\lambda_{\max} = NP$ 和 $\lambda_{\min} = 1$ 分别是最长波长和最短波长; ϵ 是一个很小的值,用于避免除数为零。

在每次迭代过程中,每个解都需要执行一次传播操作。在本文算法中,传播操作是对解中客服的数量和客服的工作时段进行更新。具体描述如下:

Step1 根据解的波长,通过改变解 X 中每个维度 d 的值,生成新解中的客服数量。为了确保每次生成的新解中客服的数量是整数,采用的策略如下:

$$1) \text{ 如果 } \lambda(X,Y) \cdot \text{rand}(-1,1) \geq 0.5, \text{ 则:} \\ X_d = X_d + \text{rand} \text{ int}(L(d) - X_d, L(d)) \quad (47)$$

$$2) \text{ 如果 } \lambda(X,Y) \cdot \text{rand}(-1,1) \leq -0.5, \text{ 则:} \\ X_d = X_d - \text{rand} \text{ int}(1, X_d) \quad (48)$$

3) 否则, x_d 保持不变。

Step2 令 $\Delta X_d = X_d' - X_d$, 如果 $\Delta X_d < 0$, 表示客服的数量要减少。这种情况下,优先去掉总服务时长最短的客服。如果 $\Delta X_d > 0$, 表示客服的数量增加,并为新增客服安排工作时段。仍然采用初始化种群的随机启发式算法生成新增客服的工作时段。

Step3 对于解中的矩阵 Y , 本文设计了一种自适应邻域搜索方法,用于改进解中每个客服的工作时段,伪代码如算法 3 所示。自适应邻域搜索通过更新邻域算子的选择概率,实现了在初期阶段鼓励全局探索,后期逐渐聚焦效果较好的邻域算子,从而提升局部开发能力。一开始,每个邻域算子的选择概率被设定为相同的。随着迭代的进行,算法会根据最近的 W 次迭代中每个算子产生新最优解的次数动态调整其选择概率。

此外,算法引入了基于波长的迭代控制策略。每个解的最大迭代次数由其波长决定,波长越长,允许的迭代次数越多,从而增强探索能力;而波长越短,迭代次数减少。每个解的迭代次数 $Iter_{\max}$ 可定义为:

$$Iter_{\text{num}} = \text{round}(\lambda/\lambda_{\max} \times Iter_{\max}) \quad (49)$$

算法 3 自适应邻域搜索算法

输入: (X, Y) , 波长 $\lambda(X, Y)$, λ_{\max} , $Iter_{\max}$

输出: 新的解 (X, Y')

1. 初始化每个邻域算子的选择概率 $p_i = 1/4$
2. 根据式(49)计算迭代次数
3. for $n=1$ to $iter_{\text{num}}$ do:
4. 使用轮盘赌法选择一个邻域算子
5. 对当前解 (X, Y) 应用选择的算子生成新解 (X, Y')
6. if $f(X, Y') < f(X, Y)$ then:
7. 替换解 $(X, Y) \leftarrow (X, Y')$
8. 记录邻域产生更优解的次数 $n_i \leftarrow n_i + 1$
9. if $n_i \bmod W = 0$:
10. 更新算子的选择概率: $p_i = \frac{n_i}{\sum_{i=1}^5 n_i}$

4.2.5 基于 Q-learning 的碎浪操作

在 WWO 算法中,碎浪操作旨在对当前最优解进行局部扰动,从而增强算法的局部搜索能力。本文在碎浪阶段引入强化学习机制,通过 Q-learning 动态选择邻域算子,以指导局部搜索的方向和策略,从而进一步提升搜索效率与解的质量[38]。Q-learning 作为一种典型的强化学习方法,适用于马尔可夫决策过程环境,能够在与环境交互中根据累积经验评估动作价值,并选择当前最优动作[39]。Q-learning 中包含 4 个元素:智能体(Agent)、环境状态(State)、动作(Action)和奖励(Reward)。Q-learning 的定义如下:

$$Q(S, A) \leftarrow (1 - \alpha)Q(S, A) + \alpha[R(S, a) + \gamma \max_{a'} Q(S', a')] \quad (50)$$

其中, α 是学习率, γ 是折扣因子, S 表示状态, A 表示每个状态中采取的动作, R 是不同动作在每个状态下获得的奖励, $Q(S, A)$ 表示当前个体的经验和记忆, $R(S, a)$ 是实际获得的经验, $R(S, a) + \gamma \max_{a'} Q(S', a')$ 表示新获的经验, $\max_{a'} Q(S', a')$ 表示基于已有经验预计在下一状态中学习到的经验。

Q-learning 可以估计每个动作(即邻域算子)在不同状态下的 Q 值,这些值可以作为在状态 S 下选择邻域算子的概率依据。

每个客服的排班被视为一个独立的智能体。在智能体的决策过程中,当前的排班被定义为状态 S , 可执行的 4 种邻域操作被视为动作 A , 动作的选择遵循 ϵ -贪心策略,即以 $1 - \epsilon$ 的概率选择当前 Q 值最大的动作,以 ϵ 概率随机选择一个动作,在选定的动作执行后的排班优化效果则作为奖励 R 。奖励可量化为目标函数的变化量 Δf 与当前最优解的目标函数值 f_{best} 的比值,即 $R = \Delta f / f_{\text{best}}$ 。奖励最大的动作就是当前要执行的最佳邻域搜索算子。

基于 Q-learning 碎浪操作的算法伪代码如算法 4 所示。

算法 4 基于 Q-learning 碎浪操作的算法

输入: 种群中的当前最优解 (X^*, Y^*) , 搜索次数 $Iter$

输出: 更优的解

1. 初始化每个状态-动作对的 Q 值为 0
2. for 1 to $Iter$ do:
3. for 每个客服:
4. if $\text{rand} < \epsilon$:
5. 随机选择一个动作

6. else;
7. 选择具有最大 Q 值的动作 $a \in A$
8. 执行选择的动作, 计算 $R = \Delta f / f_{\text{best}}$
9. 更新 Q 值
10. 生成新的解 (X^*, Y^*)
11. if $f(x^*, Y^*) < f(X^*, Y^*)$ then;
12. 则更新当前最优解为 $(X^*, Y^*) \leftarrow (X^*, Y^*)$
13. return 最优解 (X^*, Y^*)

通过这种方法, Q-learning 算法能够有效引导智能体在每个决策步骤中选择最优的邻域算子, 以优化客服排班方案。

4.2.6 随机扰动策略的折射操作

当某个解经过多次迭代仍未得到改进, 且其能量已耗尽, 波高降为 0 时, 将执行折射操作。该操作旨在引导解向当前全局最优解靠拢, 从而跳出局部最优, 避免搜索过程陷入停滞。本文在折射操作中引入了一种随机扰动策略, 使陷入停滞的解在吸收当前最优解信息的同时保留一定的随机性, 有助于在保持多样性的前提下提升找到更优解的概率。具体描述如下:

Step1 对原始解 X 中每个维度 d 上的值, 通过增加或减少一个随机数来调整, 该随机数取自当前值与最优解对应维度值之间的范围, 从而实现客服的数量向最优解的学习。

Step2 对于新变化的 ΔX_d 位客服, 如果客服的数量减少, 则优先去掉总服务时长最短的客服排班。如果客服的数量增加, 则随机分配符合约束的工作时段, 形成新解 (X', Y') 。

Step3 随机选取最优解中某个客服任意长度的工作时段, 替换当前解中同类客服等长度的工作时段, 生成新的解。

Step4 计算新解的目标函数值, 如果小于原来解的值, 则在种群中更新解。

随机扰动策略的折射操作的算法伪代码如算法 5 所示。

算法 5 随机扰动策略的折射操作的算法

输入: 种群中的当前解 (X, Y) 和最优解 (X^*, Y^*)

输出: 更新的解 (X', Y')

1. if $h_{(X, Y)} = 0$, 且当前解 (X, Y) 没有改进
2. 调整解 X 每一维中的值

$$X_d' = \left\lceil \frac{X_d + X_d^*}{2} + \text{rand}(0, 1) \frac{X_d^* - X_d}{2} \right\rceil$$
3. 计算 $\Delta X_d = X_d' - X_d$
4. if $\Delta X_d < 0$:
5. 删除解中总服务时长最短的 $|\Delta X_d|$ 行
6. if $\Delta X_d > 0$:

7. 随机分配客服的工作时段
8. 生成新的解 (X', Y')
9. for l to $\text{rand}(1, n)$ then; /* 随机执行 $1 \sim n$ 次操作 */
10. 随机选取 (X^*, Y^*) 中某个客服任意长时段 D_{rand}
11. 替换 (X', Y') 中相同位置的时段生成 (X'', Y'')
12. if $f(X'', Y'') < f(X, Y)$:
13. 更新当前解为 (X'', Y'')
14. return 当前解生成新的解

通过扰动策略对原解的结构进行适度破坏, 可有效生成新的潜在优解。

5 算法数值实验

5.1 实验用例及环境

为验证所提出算法的有效性, 本文选取了源自浙江泰隆银行宁波分行在线客服中心的个人业务类客户数据中的 8 个具有代表性的测试实例, 涵盖周末低负载、常规工作日、异常波动及高负载等几种典型的服务场景。表 5 列出了每个实例的具体特征与所属场景类型, 其中 MNFEs 表示算法在求解该实例时允许的最大目标函数评估次数, 用于控制算法的终止条件, 确保不同算法间的比较基准一致。

表 5 测试实例特征及场景类型

Table 5 Test instance features and scenario types

编号	$n_{i,r}$	场景类型	N_u	MNFEs
#1	[15,100]	周末低负载	[6,10,25]	1000
#2	[25,120]	周末低负载	[8,15,28]	1500
#3	[30,180]	正常工作日	[10,20,40]	2800
#4	[40,200]	正常工作日	[10,25,45]	3000
#5	[50,360]	客户量波动	[12,30,60]	5000
#6	[45,390]	客户量波动	[14,25,65]	6000
#7	[90,350]	高负载	[20,35,75]	8000
#8	[120,420]	高负载	[20,38,80]	10000

此外, 基于实际客户数据集, 本文进一步对各类客户的行为特征进行了统计分析, 包括客户咨询消息的平均数量 m_i 、发送每条消息所需的平均时间 μ 、愿意等待的平均时间 ΔT_i 和 $\Delta \hat{T}_i / \mu$, 以及各类客服回复每条消息的平均响应时间 θ_u 等关键变量。为了验证所采用分布假设的合理性, 本文采用最大似然估计方法对这些变量进行了指数分布拟合, 并通过 Kolmogorov-Smirnov (K-S) 检验评估其拟合优度。检验结果的统计显著性如表 6 所列, 其中 $p \geq 0.05$ 表示接受指数分布的原假设, 在表中以加粗字体显示, $p < 0.05$ 表示拒绝原假设, 即该变量在相应时段偏离指数分布, 显著性水平设定为 $\alpha = 0.05$ 。

表 6 各时段行为变量指数分布拟合检验 p

Table 6 p -values of K-S test for variables in different time slots

时段	m_1	m_2	μ	ΔT_1	ΔT_2	$\hat{\Delta T}_1$	$\hat{\Delta T}_2$	θ_1	θ_2	θ_3	通过率 %
7:00-7:30	0.234	0.189	0.312	0.456	0.598	0.267	0.398	0.512	0.289	0.445	100
7:30-8:00	0.198	0.167	0.289	0.398	0.534	0.234	0.356	0.467	0.256	0.398	100
8:00-8:30	0.156	0.134	0.267	0.356	0.487	0.198	0.312	0.421	0.034	0.356	100
8:30-9:00	0.089	0.078	0.234	0.298	0.421	0.167	0.289	0.378	0.028	0.312	100
9:00-9:30	0.034	0.089	0.045	0.067	0.134	0.045	0.134	0.187	0.023	0.234	60
9:30-10:00	0.023	0.034	0.018	0.028	0.089	0.032	0.089	0.041	0.018	0.156	40
10:00-10:30	0.028	0.045	0.034	0.078	0.123	0.067	0.123	0.034	0.025	0.189	60
10:30-11:00	0.041	0.067	0.089	0.134	0.187	0.134	0.187	0.089	0.041	0.234	80
11:00-11:30	0.187	0.156	0.234	0.298	0.356	0.234	0.298	0.356	0.289	0.398	100
11:30-12:00	0.198	0.167	0.267	0.334	0.398	0.267	0.334	0.398	0.312	0.445	100
12:00-12:30	0.234	0.198	0.289	0.356	0.421	0.298	0.378	0.445	0.356	0.489	100
12:30-13:00	0.267	0.234	0.312	0.378	0.445	0.334	0.421	0.489	0.398	0.041	80

(续表)

时段	m_1	m_2	μ	ΔT_1	ΔT_2	$\hat{\Delta T}_1$	$\hat{\Delta T}_2$	θ_1	θ_2	θ_3	通过率%
13:00-13:30	0.289	0.256	0.334	0.398	0.467	0.378	0.467	0.534	0.445	0.567	100
13:30-14:00	0.312	0.278	0.356	0.421	0.489	0.421	0.512	0.578	0.489	0.612	100
14:00-14:30	0.334	0.298	0.378	0.445	0.512	0.467	0.556	0.623	0.534	0.656	100
14:30-15:00	0.356	0.321	0.398	0.467	0.534	0.512	0.598	0.667	0.578	0.698	100
15:00-15:30	0.378	0.343	0.421	0.489	0.556	0.556	0.634	0.712	0.623	0.734	100
15:30-16:00	0.134	0.156	0.187	0.234	0.289	0.234	0.289	0.034	0.267	0.356	80
16:00-16:30	0.018	0.025	0.032	0.041	0.067	0.032	0.089	0.025	0.018	0.067	40
16:30-17:00	0.015	0.021	0.028	0.063	0.045	0.028	0.067	0.021	0.015	0.041	40
17:00-17:30	0.032	0.045	0.041	0.089	0.134	0.089	0.134	0.032	0.028	0.123	60
17:30-18:00	0.041	0.067	0.089	0.123	0.187	0.134	0.187	0.089	0.041	0.189	80
18:00-18:30	0.156	0.134	0.189	0.234	0.289	0.234	0.289	0.356	0.267	0.398	100
18:30-19:00	0.189	0.167	0.234	0.278	0.334	0.278	0.334	0.421	0.312	0.467	100
19:00-19:30	0.234	0.198	0.267	0.312	0.378	0.312	0.378	0.489	0.356	0.534	100
19:30-20:00	0.267	0.234	0.298	0.356	0.421	0.356	0.421	0.556	0.398	0.598	100
20:00-20:30	0.298	0.267	0.334	0.398	0.467	0.398	0.467	0.623	0.445	0.656	100
20:30-21:00	0.334	0.298	0.378	0.445	0.512	0.445	0.512	0.689	0.489	0.712	100
21:00-21:30	0.378	0.343	0.421	0.489	0.556	0.489	0.556	0.734	0.534	0.756	100
21:30-22:00	0.421	0.378	0.467	0.534	0.598	0.534	0.598	0.778	0.578	0.798	100

结果显示,在300个变量-时段组合中,共有252个组合通过检验,整体通过率达84%,表明大部分变量与指数分布假设具有良好的统计拟合效果。从变量来看,等待时间相关的变量 ΔT_i 和 $\hat{\Delta T}_i$ 的拟合效果最佳,客户咨询消息的数量 m_i 在高峰时段容易出现偏离,通过率相对较低。这表明客户的主观行为比系统响应时间具有更强的不确定性。整体上来说,检验结果揭示了客服调度中在高峰时段需求复杂多变,而在非高峰时段具有相对稳定的运营规律,这为后续的预测建模和调度策略制定提供了重要的理论依据。

各类行为变量的统计值如表7所列。

表7 部分参数的评估值

Table 7 Estimated value of some parameters

参数	取值
m_i	[3,15]
μ	[30,200]
ΔT_i	[240,520]
θ_u	[9,24]
$\hat{\Delta T}_i$	[60,200]

输入变量中的一些参数具有确定的取值,如表8所列。其中, c_i 根据人均年收入水平计算, α_i 根据客服的工资估算。

表8 输入变量取值

Table 8 Values of input variables

参数	取值
c_i /(元/分钟/人次)	[0.5,1]
s_u /时	$s_3=24, s_2=30, s_1=35$
α_i	[480,560]
T	30
T^{\min}	12
\hat{T}	5
T^{\max}	16

为评估QWWO算法的性能,本文选取了若干高效的元启发式算法作为对比方法。其中部分算法虽原用于如护士排班等其他人员调度问题,但其核心优化机制具有良好的通用性,可适用于本文所研究的在线客服调度问题。

为确保对比算法在当前问题场景下的适应性,本文在不改变算法核心机制的前提下,结合客服调度问题的特定约束,对其编码方式、适应度函数及解空间处理方式等进行了适当的结构性调整,使其能够有效求解本文的问题。

为了公平地进行比较,所有算法均采用以往研究中最常见或推荐的基本配置,并在不同的参数设置下进行测试,最终

选取在测试用例上平均性能最优的参数设置进行对比分析。各算法及其控制参数的调整如下:

1)遗传算法(Genetic Algorithm,GA)^[37]:种群大小 $N=40$,变异率 $\rho_m=0.2$,交叉率 $\rho_c=1$ 。

2)自适应遗传算法(Adaptive Genetic Algorithm,AGA)^[38]:种群大小 $N=40$,交叉率 $\rho_c=1$,变异率 $\rho_m=0.8$,基本利用率 $r=0.04$ 。

3)粒子群算法(Particle Swarm Optimization,PSO)^[39]:种群大小 $N=30$,学习因子 $c_1=c_2=1.0$,最大惯性权重值 $\omega_{\max}=0.9$,最小惯性权重值 $\omega_{\min}=0.4$ 。

4)改进的粒子群算法(Advanced Particle Swarm Optimization,APSO)^[40]:种群大小 $N=30$,学习因子 $c_1=c_2=1.5$,最大速率 $v_{\max}=10$;

5)模拟退火算法(Simulated Annealing,SA)^[41]:种群大小 $N=50$,初始温度 $T_0=18.64$,冷却速度 $\alpha=0.99$,终止温度 $T_f=0.87$,交换率 $\sigma=0.11$,接受移动比率 $\rho=0.08$ 。

6)差分进化算法(Differential Evolution,DE)^[42]:种群大小 $N=50$,交叉概率 $c_r=0.9$,变异因子 $F=0.5$ 。

7)自适应差分进化算法(Self-Adaptive Differential Evolution,SADE)^[43]:种群大小 $N=50$,交叉概率 $c_r=0.9$,变异因子 $F=0.5$,自适应控制参数概率 $\tau=0.1$ 。

8)非混合Q-learning水波优化(WWO)算法:种群大小 $N=40, k_{\max}=12$,碎浪操作随机选择邻域算子。

9)混合Q-learning水波优化(QWWO)算法:种群大小 $N=40, k_{\max}=12$,Q-learning算法中的参数,学习率 $\alpha=0.1$,折扣因子 $\gamma=0.9$,探索概率 $\epsilon=0.2$,逐渐衰减至一个较小的值0.01。

所有算法均采用统一的随机初始化策略,并在相同的约束条件下运行。终止条件统一为目标函数的评估次数,如表5最后一列MNFs中所示。另外,考虑到问题参数的随机性,为了充分测试算法在各种不确定场景下的性能,对于每个用例的场景,每种算法在每个实例上使用随机种子分别独立运行50次,并记录在这50次运行中获得的适应度值的最佳(Minimum)、最差(Maximum)、中位数(Median)以及标准差(Std)。最终的适应度评价基于每个用例下所有场景的平均值计算。

5.2 实验结果及分析

图6-图13展示了9种算法在测试用例上运行结果的

箱线图,图中包括各算法在8个实例上得到的目标函数值的均值(用绿色星形表示)、中位数、最大值、最小值、第一个四分位数(Q_1)和第三个四分位数(Q_3)。任何目标值低于下限 $Q_1 - 1.5(Q_3 - Q_1)$ 或高于上限 $Q_3 + 1.5(Q_3 - Q_1)$ 都被视为异常值。对本文算法和其他算法在每个实例上得到的结果进行非参数的 Wilcoxon 秩和检验,并将所得 p 值标注在相应箱线图的最大值上方。若 p 值小于 0.05,则在其前添加红色“+”标记,以表明在 95%置信水平下存在显著统计差异。

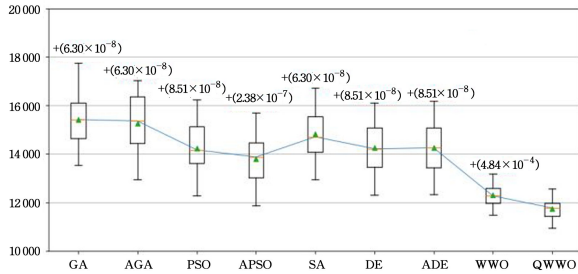


图 6 9种算法在测试用例 #1 上的箱线图(电子版为彩图)
Fig. 6 Box plots of the results obtained by the 9 comparative algorithms on instance #1

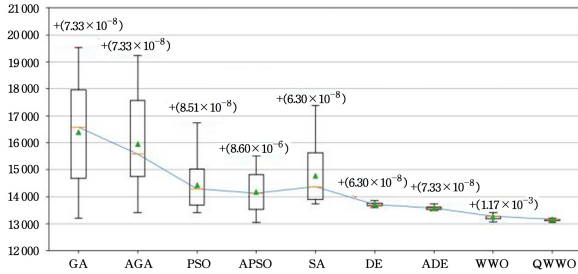


图 7 9种算法在测试用例 #2 上的箱线图(电子版为彩图)
Fig. 7 Box plots of the results obtained by the 9 comparative algorithms on instance #2

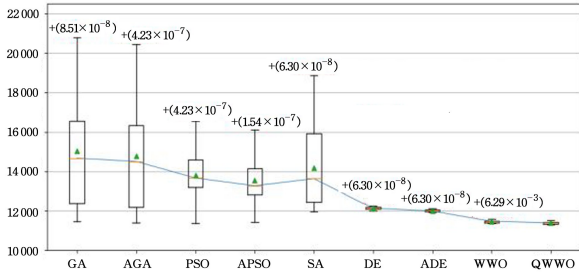


图 8 9种算法在测试用例 #3 上的箱线图(电子版为彩图)
Fig. 8 Box plots of the results obtained by the 9 comparative algorithms on instance #3

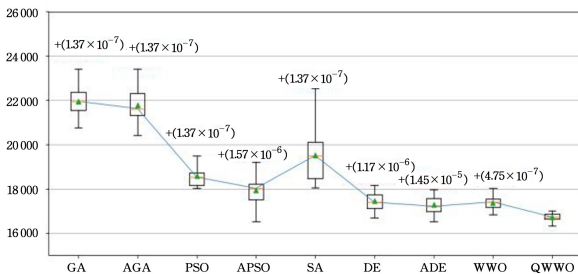


图 9 9种算法在测试用例 #4 上的箱线图(电子版为彩图)
Fig. 9 Box plots of the results obtained by the 9 comparative algorithms on instance #4

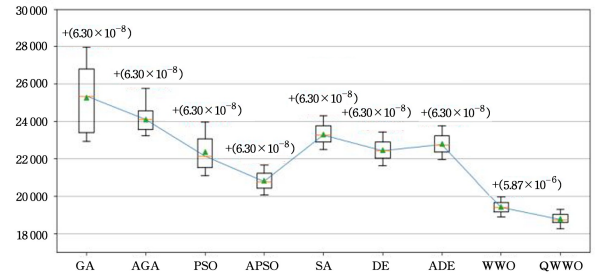


图 10 9种算法在测试用例 #5 上的箱线图(电子版为彩图)
Fig. 10 Box plots of the results obtained by the 9 comparative algorithms on instance #5

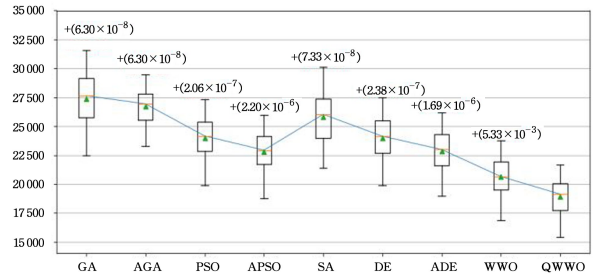


图 11 9种算法在测试用例 #6 上的箱线图(电子版为彩图)
Fig. 11 Box plots of the results obtained by the 9 comparative algorithms on instance #6

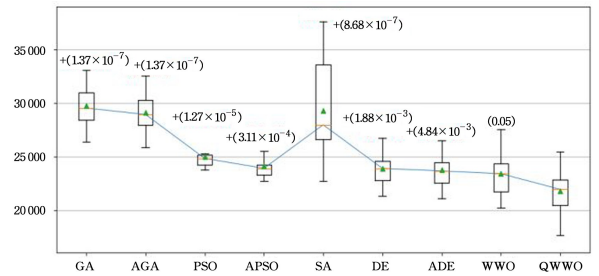


图 12 9种算法在测试用例 #7 上的箱线图(电子版为彩图)
Fig. 12 Box plots of the results obtained by the 9 comparative algorithms on instance #7

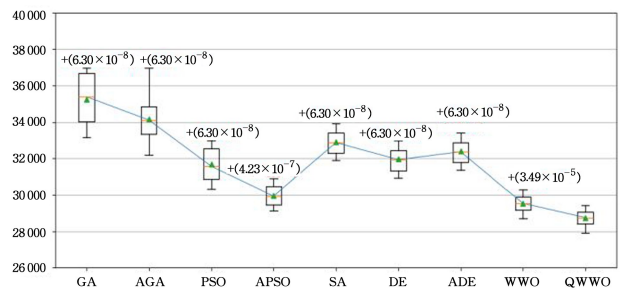


图 13 9种算法在测试用例 #8 上的箱线图(电子版为彩图)
Fig. 13 Box plots of the results obtained by the 9 comparative algorithms on instance #8

在这 8 个测试实例中,所提出的 QWWO 在 8 种对比算法中均获得了最优的中位目标值。根据统计检验结果,在实例 #7 中, QWWO 和 WWO 的结果没有显著差异,在其余所有实例上, QWWO 的性能都显著优于对比算法。QWWO 通过强化学习(Q-learning)动态优化邻域选择,展现出更强的局部搜索能力,有效提升了解的精度和稳定性。相比之下, GA 的表现最差,主要原因是其全局搜索能力较

弱,难以跳出局部最优。

在小规模实例#1和#2中,PSO,APSO,DE和ADE的表现相对接近,说明其搜索机制在平衡全局与局部搜索方面具有一定优势。随着问题复杂度略有上升,GA,AGA,PSO,APSO和SA的性能明显下降,反映出其对客户到达波动的适应性较差,尤其是AGA和APSO的自适应机制在处理动态波动时调整能力有限。相比之下,DE和ADE的差分进化策略在波动环境下表现出更强的稳定性与解的质量。

在波动较大的实例#5和#6中,PSO,SA,DE和ADE的性能均处于中等水平,表明在某些波动模式下其适应性存在局限。APSO在实例#6中的中位值优于PSO,而在实例#5中二者的中位值相近,说明APSO的自适应参数机制在某些场景下可提升算法表现。

在大规模高负载的实例#7和#8中,PSO,SA,DE和ADE的表现仍维持在中等水平,显示出一定的适应性,但不足以充分应对高强度负载的调度需求。在实例#8中,APSO的表现优于PSO,尤其在负载特征明显的场景下,其参数

调整能力更能匹配实际需求。

总体来看,在小规模、波动较小的场景下,GA,PSO,SA和DE等元启发式算法能够有效平衡搜索效率与收敛速度,取得较好性能;AGA,APSO和ADE通过引入自适应机制增强了局部搜索能力,在中等规模、波动适中的实例中表现更优,适用于常规工作日等典型客服调度场景。然而,在客户需求波动剧烈或高负载的复杂环境下,这些算法的局限性逐渐显现,主要体现在搜索精度和收敛稳定性方面。

相比之下,融合邻域搜索策略的混合启发式算法,如WVO和本文QWWO,展现出更强的优化能力。QWWO在WVO的基础上引入强化学习机制,动态调整邻域操作策略,显著提升了了解的质量与适应性,在复杂场景中表现出优越的调度性能。

为确保公平性,所有算法均使用相同的目标函数评估次数作为终止条件,但不同算法的运行时间仍存在细微差异,如表9所列。

表9 算法在测试实例上消耗的中位数CPU时间

Table 9 Median CPU time consumed by each comparative algorithm on the test instances

编号	GA	PSO	APSO	SA	DE	ADE	WVO	QWWO
#1	0.32	0.27	0.31	0.39	0.29	0.30	0.31	0.34
#2	0.41	0.34	0.38	0.48	0.37	0.40	0.38	0.42
#3	0.68	0.58	0.65	0.72	0.52	0.62	0.67	0.69
#4	0.85	0.70	0.75	0.79	0.64	0.76	0.82	0.89
#5	1.39	1.20	1.54	1.62	1.36	1.30	1.35	1.43
#6	1.92	1.76	1.89	2.12	1.65	1.81	1.93	1.97
#7	12.49	11.89	12.62	13.24	12.31	12.14	12.47	12.56
#8	16.34	15.73	16.16	17.73	15.96	16.30	16.48	16.84

从表9中可以看出,在9种算法中,PSO与DE耗时最短,原因在于其框架相对简单且搜索算子计算高效。WVO算法虽比PSO和DE耗时更长,但在多数情况下短于其余算法。相比之下,QWWO需额外耗费约3%的时间执行邻域搜索算子和Q-learning计算过程。考虑到最终解的质量显著提升,这些额外时间成本微不足道。

6 实证分析

为验证所提模型与算法的实际应用效果,本文基于浙江泰隆银行在线客服中心的真实运营数据开展实证分析。

6.1 不确定参数预测

浙江泰隆银行在线客服中心支持4类业务,包括个人业务、公司业务、信用卡业务和电子银行业务。本文聚焦于个人业务中的客户到达数据,分析其到达模式。通过多次迭代收集和筛选,获得了银行来访客户的原始数据集,包括客户类型、到达时间、服务客服、对话持续时长、业务类别、对话结束方、交互状态、客户响应时长、客服响应时长、服务结束时间、客户评级、客户评价,以及客户使用的操作系统和浏览器等特征信息。为保护数据隐私和安全性,原始数据集不包含具体的客户个人信息。

所使用的数据集涵盖2023年2月7日至2023年5月7日连续3个月内客户到达的原始数据。为便于分析,原始数据中的客户到达的时间戳被转换为均匀分布的时间序列

格式。将每日15小时的服务时段划分为30个以30分钟为间隔的时间段,并将客户到达数据聚合至对应的时间区间内,从而生成客户到达的频率分布。基于上述转换,本文对2023年2月7日到2023年5月7日连续3个月内客户到达的数据进行统计分析,得出客户到达的频率分布特征,结果如图14所示。从图中可以看出,在整个考虑的时间段内,客户的到达量在整体上呈现出相对稳定的变化趋势。

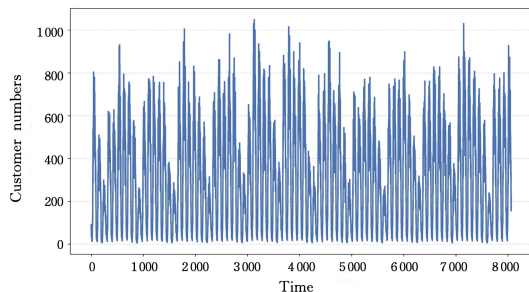


图14 客户到达频率分布图

Fig. 14 Customer arrival frequency distribution

图15展示了2023年4月21日至27日(周一至周日)期间每日客户到达的分布情况。从图中可以看出,客户到达量存在明显的周期性波动。周一为全周最繁忙的一天,尤其是在上午时段,客户到达量达到峰值。其余工作日的到达模式较为相似,通常在上午和下午各出现一个高峰,尽管整体到达量略低于周一,但波动趋势基本一致。相比之下,周末的客户到

达量显著下降,反映出客户咨询需求在非工作日明显减弱。

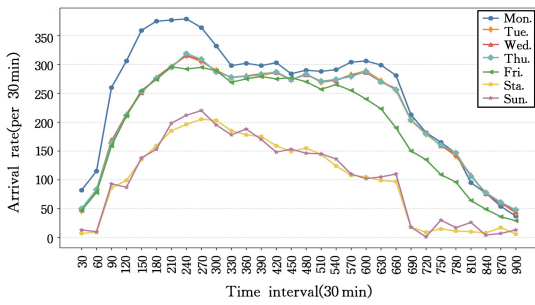


图 15 周一到周日客户到达频率分布图

Fig. 15 Customer arrival frequency distribution in a week

预测模型训练阶段使用 2023 年 2 月 27 日至 2023 年 5 月 7 日的客户到达数据,以充分捕捉客户到达的周期性规律与趋势性特征。随后,基于每日新增数据对模型进行持续更新与再训练,以适应客户需求的短期波动,提升客户到达率的动态预测能力。在训练阶段,LSTM 的时间复杂度可表示为 $O(E \cdot N \cdot T \cdot d^2)$,其中 E 为训练轮数, N 为样本数, T 为输入时间步长度, d 为 LSTM 单元维度。本文使用的数据集覆盖 90 天、每半小时记录一次,共计约 4 320 条时间步,采用滑动窗口构造训练样本。以 $T=15, d=64$ 为例,模型训练复杂度在 10^{10} 数量级。

利用 LSTM 模型对 2023 年 4 月 21 日到 27 日这一周的客户到达量进行预测,预测阶段的复杂度是 $O(T \cdot d^2)$ 。图 16 显示了 2023 年 4 月 21 日到 27 日这 7 天客户到达的真实分布与预测分布对比情况。可以看出,LSTM 模型的预测结果与实际到达量吻合度较高,预测值能够较准确地反映客户到达的波动趋势。

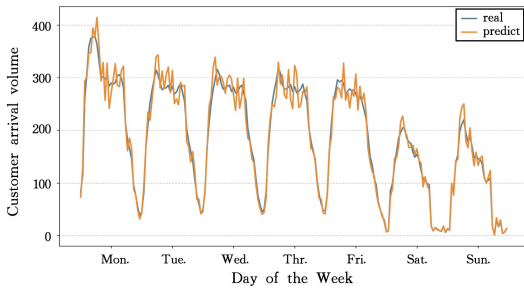


图 16 客户实际到达分布与预测分布

Fig. 16 Real and predicted customer arrival frequency distribution

6.2 算法的有效性

为了验证本文算法 QWWO 的有效性,将其与以下 3 种方法的运行效果进行了对比分析。

1) 班次固定的原调度方法。该方法为银行当前采用的调度方案,班次和工作时段均为固定预设,客服的数量根据历史经验配置。

2) 班次固定的客服配置调度。在固定班次不变的前提下,根据 LSTM 模型预测的客户到达量确定每个班次所需的客服人数。对应于式(33),在决策变量 Y 已知的情况下,以客服人员配置 X 为优化目标。采用文献[44]提出的改进水波优化算法 AWWO 进行求解。

3) 固定客服数量的调度。在客服的总人数固定的前提下,打破传统固定班次的约束,允许客服的工作时段在满足劳

动法和工作时长约束的前提下灵活安排。此时,式(33)中 Y 为优化变量,目标为生成最优排班方案。采用遗传算法[40]来求解客服的最优排班方案。

针对 2023 年 4 月 21 日到 27 日这 7 天的数据实例,图 17 展示了 4 种方法所获取的目标函数值(即总成本)以及 LSTM 预测方法的准确率,其中折线图分别呈现了 4 种方法在这 7 天实例上的目标函数值,柱状图则显示了 LSTM 预测方法的准确率。在周一到周三的高峰期,本文调度方案的目标函数值与前三种方案相比分别降低约 20.7%,15.6% 和 13.8%,表明其在高负载时性能更优;在周六和周日客户量较小时,目标函数值分别降低约 17.7%,14% 和 10.2%;在这一周实例上的平均目标函数值分别降低约 18.6%,17.2% 和 14.1%。

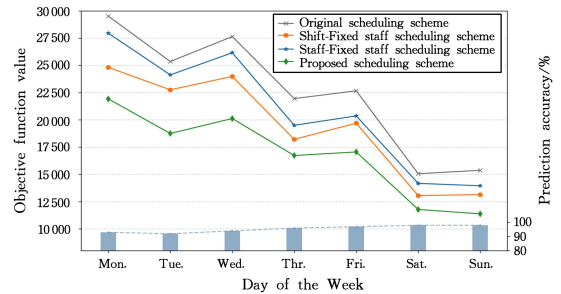


图 17 4 种方法和预测准确率在实例上的对比

Fig. 17 Comparison of the four methods and the prediction accuracy of LSTM

结果表明,作为基准的原银行调度方案在每日实例中的目标函数值均最高,反映出其在资源配置与服务效率方面存在明显改进空间。相比之下,固定客服人数下的灵活排班方案通过调整每位客服的工作时段,在一定程度上优化了资源分配,目标函数值较低。然而,客服的总人数固定,当客户需求激增(如周一高峰)时,服务能力受限,导致等待时间延长,增加了等待成本。

固定班次下的动态人员配置方案则依据客户预测结果调整每个班次的客服人数,整体性能优于固定人数方案,表明动态调整人员配置有助于缓解需求波动的影响。但由于班次安排固定,仍可能出现高峰期人力不足或低谷期浪费的问题。

相比之下,本文 QWWO 双重优化调度方案在 4 种方法中表现最优,目标函数值始终最低。该方案同时优化了客服的数量和工作时段,显著增强了对客户需求波动的适应能力,避免了资源冗余或缺缺问题,提升了人力资源利用效率,并有效缩短了客户等待时间,提升了服务质量与客户满意度。

从连续 7 天的测试结果看出,QWWO 所实现的目标函数值在每个实例中均为最低,明显优于原有银行方案、固定人数方案及固定班次方案,验证了其在复杂客服调度问题中的优越性与实用性。LSTM 预测方法的准确率都在 90% 以上,进一步为该方案提供了可靠支持。

以 2023 年 4 月 21 日为例,进一步分析客服中心的调度方案。为实现全时段服务覆盖,原始排班方案设置了早班、晚班和两段班 3 种班种,每个班种设若干错峰安排的班次,每个班次由两个连续时间段构成,以确保客服在工作时段内具备稳定的响应能力。

根据业务类别和技能熟练度,人工坐席被划分为多个服

务群组,每组人数从数人至十余人不等,并根据服务需求分配至不同班次,以满足各类客户的服务要求。

在该日的调度方案中,个人业务类别的在线客服的总人数为 60 人,每个班次的客服人数分配如图 18 所示。

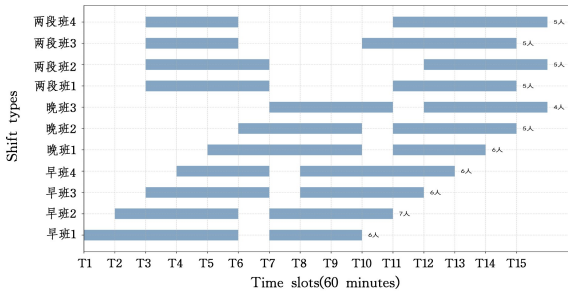


图 18 原调度方案中客服人数及班次分配情况

Fig. 18 Staffing levels and shift assignments in current customer service operations

以高级客服为例,通过应用本文调度方法,其工作时段安排如图 19 所示。从图中可以看出,客服的工作时段安排比较灵活,在客户需求高峰期集中安排任务,在低谷期设置充分的休息时段,从而提升服务的响应效率。

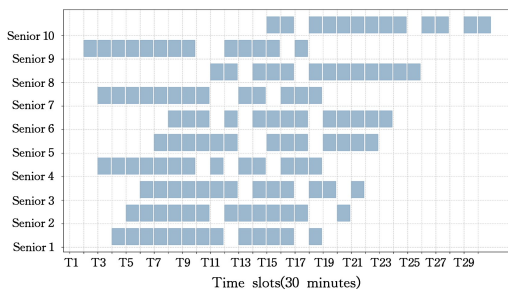


图 19 调度后高级客服的人数及工作时段分配情况

Fig. 19 Senior staff levels and shift assignments after scheduling

应用本文方法后,客服中心的资源配置得到显著优化,所需的客服数量由原来的 60 人减少到 48 人,其中高级客服减少 2 人,中级客服减少 4 人,初级客服减少 6 人。客服总数减少 20%,高级、中级和初级客服的减少比例较为均衡,表明该调度方法在保证服务质量的同时,显著降低了人力成本,也反映了方法在不同技能层级的客服分配上的合理性。调度前后各时段的客服人数安排如图 20 所示。

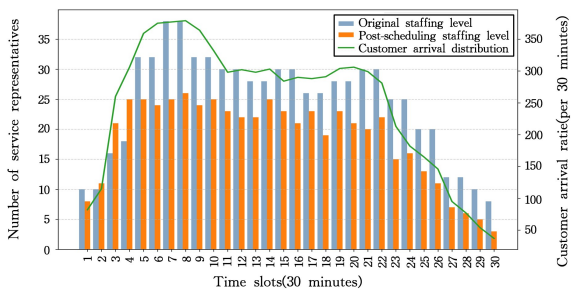


图 20 调度前后各时段客服人数对比

Fig. 20 Comparison of agent allocation across time slots before and after optimization

6.3 预测敏感度

为了更全面地探讨 LSTM 模型预测的准确性与客服调度效果之间的关系,本文进一步在 2023 年 4 月 21 日到 27 日

这 7 天的数据实例上进行了模拟实验,分别模拟了需求预测准确率在 50%,60%,70%,75%,80%,85%,87.5%,90%,92.5%,95%,97.5%,98.5%时的客户到达场景,并应用本文方法求解不同预测精度下的模拟实例。图 21 展示了不同场景下的目标函数值。

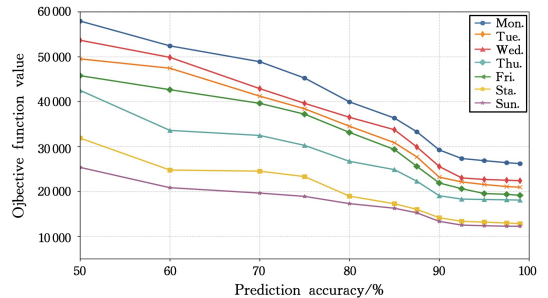


图 21 不同预测精确度下的目标函数值

Fig. 21 Objective function values under different (simulated) prediction accuracies

从图 20 中可以看出,当预测准确度仅为 50%时,目标函数值(即总成本)显著偏高,是预测准确度为 90%时的 2~3 倍。这表明,在预测精度较低的情况下,基于预测需求进行的客服调度对降低成本和提升客户满意度的帮助极为有限,人员配置和排班的优化效果大打折扣。随着预测准确度的逐步提高,如从 50%增加至更高水平,目标函数值呈现出稳步下降的趋势。更准确的客户需求预测使得客服配置和工作时段能够更好地匹配实际需求,从而显著提升交付效率。当预测准确度超过 90%时,目标函数值的下降速度放缓。原因在于,在客户需求和客服匹配时,客户的分配方案可以有效处理客户需求的少量波动。这种鲁棒性使得进一步提升预测准确性对客服调度效果的增益变得有限,性能趋于稳定。这一分析不仅验证了预测准确性对客服调度性能的重要性,还为实际应用中如何平衡预测模型的复杂性与调度效率提供了理论依据和实践指导。

结束语 本文针对银行客服中心在线客服的人员配置和排班调度问题,构建基于混合整数线性规划的优化模型,并提出了一种混合机器学习预测与启发式优化的方法。为验证模型和方法的性能,采用实际问题实例进行测试。实验结果表明,相较于对比方法,本文方法在降低成本和提升调度效率方面具有显著优势。此外,为了探究预测准确性和客服调度性能之间的关系,本文通过模拟不同预测精度下的客户到达量,分析其对调度效果的影响。实验结果表明,当预测精度较低时,由于客户到达量的不确定性增加,调度方案的成本显著上升,客户等待时间延长。当预测准确率超过 90%时,客户等待时间和运营成本下降趋于平缓。

未来研究可进一步调整模型和算法的输入特征、服务规则和约束条件,深化模型设计与优化,并结合真实数据开展验证,以实现向公司业务、信用卡业务、电子银行业务等更复杂服务环境的有效迁移。

在方法上,下一步研究考虑将预测与调度两个环节进一步整合为一个统一的优化过程,采用端到端(End-to-End)学习策略,以提升模型的泛化能力,以及调度方案的实时性和整体性能。

参考文献

- [1] DAI T, ZHAO X. A queuing model for 1-n interactive service mode in online customer service systems[J]. *Journal of Systems & Management*, 2021, 30(3): 562-570.
- [2] YU M, ZHAO Y, CHANG C G, et al. Fluid models for customer service web chat systems with interactive automated service[J]. *Flexible Services and Manufacturing Journal*, 2023, 35(2): 572-598.
- [3] MORTEZA S, OMAR K H, ELIZABETH C. Past, present and future of contact centers: a literature review [J]. *Business Process Management Journal*, 2017, 23(3): 574-597.
- [4] DAI T, YU T T, ZHAO X. Decision strategy of single server online customer service with variable service rates[C]// 2019 International Conference on Industrial Engineering and Systems Management (IESM). IEEE, 2019: 1-6.
- [5] SU Q, ZHAO F. Research on staff scheduling algorithm for multi-skill call center[J]. *Systems Engineering-Theory & Practice*, 2015, 35(1): 239-246.
- [6] ERNST A T, JIANG H, KRISHNAMOORTHY M, et al. Staff scheduling and rostering: A review of applications, methods and models[J]. *European Journal of Operational Research*, 2004, 153(1): 3-27.
- [7] CHAN W, ANH T T, L'ECUYER P, et al. Two-stage chance-constrained staffing with agent recourse for multi-skill call centers[C]// 2016 Winter Simulation Conference (WSC). IEEE, 2016: 3189-3200.
- [8] ROBBINS T R, HARRISON T P. A stochastic programming model for scheduling call centers with global Service Level Agreements[J]. *European Journal of Operational Research*, 2010, 207(3): 1608-1619.
- [9] TÜRKER T, DEMIRIZ A A. An MIP model to schedule the call center workforce and organize the breaks[C]// Innovations Through Mathematical And Statistical Research: Proceedings of the 2nd International Conference on Mathematical Sciences and Statistics (ICMSS2016). 2016: 020048.
- [10] ÁLVAREZ E, FERRER J C, MUÑOZ J C, et al. Efficient shift scheduling with multiple breaks for full-time employees: A retail industry case[J]. *Computers & Industrial Engineering*, 2020, 150: 106884.
- [11] VASILYEV I, USHAKOV A V, ARKHIPOV D, et al. Fast heuristics for a staff scheduling problem with time interval demand coverage[M]// *Mathematical Optimization Theory and Operations Research*. Cham: Springer, 2024: 212-228.
- [12] WU X Q, ZHOU S H. Sequencing and scheduling appointments on multiple servers with stochastic service durations and customer arrivals[J]. *Omega*, 2022, 106: 102523.
- [13] PANDEY P, GAJJAR H, SHAH B J. Determining optimal workforce size and schedule at the retail store considering over-staffing and understaffing costs[J]. *Computers & Industrial Engineering*, 2021, 161: 107656.
- [14] MA Y, LIU L L. Solving call center agent scheduling problem through improved adaptive genetic algorithm[C]// 2012 Fifth International Symposium on Computational Intelligence and Design. 2012: 27-30.
- [15] YAN L J, KONG X M, WANG N, et al. Research on artificial seat scheduling algorithm based on genetic algorithm[C]// Proceedings of the 2024 8th International Conference on Control Engineering and Artificial Intelligence. 2024: 26-29.
- [16] REKIK M, CORDEAU J F, SOUMIS F. Implicit shift scheduling with multiple breaks and work stretch duration restrictions [J]. *Journal of Scheduling*, 2010, 13(1): 49-75.
- [17] WANG H B, ALIDAE B, ORTIZ J, et al. The multi-skilled multi-period workforce assignment problem [J]. *International Journal of Production Research*, 2021, 59(18): 5477-5494.
- [18] NTHIMO L, MOKOENA T, MODUPE A, et al. Call centre shift schedule optimisation using local search heuristics[C]// 2021 IEEE Africon. IEEE, 2021: 1-6.
- [19] DHIBAR S, JAIN M. Metaheuristic optimization and strategic behavior of Markovian vacation queue with retrial policy: application to virtual call center[J]. *Evolutionary Intelligence*, 2025, 18(1): 13.
- [20] XU Y, WANG X L. An artificial bee colony algorithm for scheduling call centres with weekend-off fairness[J]. *Applied Soft Computing*, 2021, 109: 107542.
- [21] NIU G, VADIVELLO J, XU M N. Agent-based queuing model for call center forecasting and management optimization[M]// *Advances in Business and Management Forecasting*. Emerald Publishing Limited, 2019: 121-133.
- [22] DE MOURA A F, DE ARAÚJO C M, NAPOLITANO D M R, et al. Optimization of operational costs of Call centers employing classification techniques [J]. *Research, Society and Development*, 2020, 9(11): e86691110491.
- [23] HENRY C, VISHWA K, DAVID D, et al. Forecasting call center arrivals using temporal memory networks and gradient boosting algorithm [J]. *Expert Systems with Applications*, 2023, 224: 119983.
- [24] NING C, YOU F Q. Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming[J]. *Computers & Chemical Engineering*, 2019, 125: 434-448.
- [25] ZHENG Y J, XIE X C, ZHANG Z Y, et al. Deep reinforcement learning assisted memetic scheduling of drones for railway catenary deicing[J]. *Swarm and Evolutionary Computation*, 2024, 91: 101719.
- [26] XU Y, LI X B, MENG X P. A Q-learning based iterated local search algorithm for human-UAV cooperation in restoring transmission network[J]. *Expert Systems with Applications*, 2024, 252: 124200.
- [27] RAKSHIT P, KONAR A, BHOWMIK P, et al. Realization of an Adaptive Memetic Algorithm Using Differential Evolution and Q-Learning: A Case Study in Multirobot Path Planning [J]. *IEEE Transactions on Systems, Man, and Cybernetics, Systems*, 2013, 43(4): 814-831.
- [28] JOUINI O, POT A, KOOLE G, et al. Online scheduling policies for multiclass call centers with impatient customers[J]. *European Journal of Operational Research*, 2010, 207(1): 258-268.
- [29] IBRAHIM R, YE H, L'ECUYER P, et al. Modeling and forecasting call center arrivals: A literature survey and a case study [J]. *International Journal of Forecasting*, 2016, 32(3): 865-874.

- [30] SHEN H P, HUANG J H. Interday Forecasting and Intraday Updating of Call Center Arrivals[J]. *Manufacturing & Service Operations Management*, 2008, 10(3): 391-410.
- [31] BARROW D, KOURENTZES N. The impact of special days in call arrivals forecasting: A neural network approach to modelling special days [J]. *European Journal of Operational Research*, 2018, 264(3): 967-977.
- [32] MANNO A, ROSSI F, SMRIGLIO S, et al Comparing deep and shallow neural networks in forecasting call center arrivals[J]. *Soft Computing*, 2023, 27(18): 12943-12957.
- [33] HU Y, CATO K D, CHAN C W, et al. Use of real-time information to predict future arrivals in the emergency department[J]. *Annals of Emergency Medicine*, 2023, 81(6): 728-737.
- [34] HUBER J, STUCKENSCHMIDT H. Daily retail demand forecasting using machine learning with emphasis on calendric special days[J]. *International Journal of Forecasting*, 2020, 36(4): 1420-1438.
- [35] HUANG R J, WEI C J, WANG B H, et al. Well performance prediction based on Long Short-Term Memory(LSTM) neural network[J]. *Journal of Petroleum Science and Engineering*, 2022, 208: 109686.
- [36] ZHENG Y J, LU X Q, DU Y C, et al. Water wave optimization for combinatorial optimization: Design strategies and applications[J]. *Applied Soft Computing*, 2019, 83: 105611.
- [37] SARDJONO W, PRIATNA W, NUGROHO D S, et al. Genetic algorithm implementation for application of shifting work scheduling system[M]. ICIC International Association, 2021.
- [38] ZHANG L, WANG L, ZHENG D Z. An adaptive genetic algorithm with multiple operators for flowshop scheduling[J]. *The International Journal of Advanced Manufacturing Technology*, 2006, 27(5/6): 580-587.
- [39] AKJIRATIKARL C, YENRADEE P, DRAKE P R. PSO-based algorithm for home care worker scheduling in the UK[J]. *Computers & Industrial Engineering*, 2007, 53(4): 559-583.
- [40] AKJIRATIKARL C, YENRADEE P, DRAKE P R. An Improved Particle Swarm Optimization Algorithm for Care Worker Scheduling[J]. *Industrial Engineering & Management Systems*, 2008, 7(2): 171-181.
- [41] ZANAZZO E, CESCHIA S, DOVIER A, et al. Solving the medical student scheduling problem using simulated annealing[J]. *Journal of Scheduling*, 2024, 28: 233-246.
- [42] SHAHNAZARI P, TAVAKKOLI-MOGHADDAM R, AZARKISH M, et al. A differential evolution algorithm developed for a nurse scheduling problem[J]. *The South African Journal of Industrial Engineering*, 2012, 23(3): 68.
- [43] PUNYAKUM V, NITISIRI K, SETHANAN K, et al. A self-adaptive differential evolution for the technician routing and scheduling problem[J]. *Engineering and Applied Science Research*, 2023, 50(3): 262-269.
- [44] LU X Q, WU C X, YANG X H, et al. Adapted water wave optimization for integrated bank customer service representative scheduling[J]. *International Journal of Production Research*, 2023, 61(1): 320-335.



LU Xueqin, born in 1980, Ph.D candidate, associate professor, is a member of CCF(No. 88689G). Her main research interest includes intelligent optimization algorithms and applications.



LIU Yangguang, born in 1975, Ph.D, professor, master's supervisor, Zhejiang Province 151 Talent Program awardee, is a member of CCF(No. 20163M). His main research interests include multi-scale data intelligence, AI and applied optimization.

(责任编辑:何杨)