

# 控制与决策

Control and Decision

多结构型任务驱动的上门服务调度优化模型与自适应大规模邻域搜索算法

展月, 姜兆勤, 刘振元

引用本文:

展月, 姜兆勤, 刘振元. 多结构型任务驱动的上门服务调度优化模型与自适应大规模邻域搜索算法[J]. *控制与决策*, 2024, 39(3): 947–955.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2022.1281>

---

## 您可能感兴趣的其他文章

### Articles you may be interested in

[基于双种群模糊引力搜索算法的舰载机甲板作业调度](#)

Flight deck operations scheduling based on dual population fuzzy gravitational search algorithm  
*控制与决策*. 2021, 36(11): 2751–2759 <https://doi.org/10.13195/j.kzyjc.2020.0523>

[基于粒子群算法的满载需求可拆分车辆路径规划](#)

Split vehicle route planning with full load demand based on particle swarm optimization  
*控制与决策*. 2021, 36(6): 1397–1406 <https://doi.org/10.13195/j.kzyjc.2019.1323>

[铁路集装箱中心站资源分配与作业调度联合优化](#)

Integrating optimization of resource allocation and handling scheduling in railway container terminal  
*控制与决策*. 2021, 36(12): 3063–3073 <https://doi.org/10.13195/j.kzyjc.2020.0597>

[考虑卸载顺序约束的成品油二次配送车辆路径问题](#)

Vehicle routing problem of refined oil secondary distribution considering unloading sequence constraints  
*控制与决策*. 2020, 35(12): 2999–3005 <https://doi.org/10.13195/j.kzyjc.2018.1756>

[复合类别航站楼分配问题的改进和声搜索算法](#)

Solving composite airport gate allocation problem with improved harmony search  
*控制与决策*. 2020, 35(11): 2743–2751 <https://doi.org/10.13195/j.kzyjc.2019.0242>

# 多结构型任务驱动的上门服务调度优化模型与 自适应大规模邻域搜索算法

展月, 姜兆勤, 刘振元<sup>†</sup>

(华中科技大学 人工智能与自动化学院, 武汉 430074)

**摘要:** 研究一类特殊的上门服务调度问题, 该问题具有一般上门服务调度问题的要求, 即要求可能具备不同技能水平的服务人员从同一站点出发, 按路径执行被分配的任务后返回站点. 被分配的任务在已有研究中对对应图内点, 而现实中的上门服务任务还可能具备内部结构(称为多结构型的任务), 因此, 在这类问题中路径的生成过程无法由任务序列本身确定, 需要考虑任务指派和含出入点选取的路径规划的协同优化. 通过分析此类问题特征, 建立以总拖期最小化为目标的混合整数规划模型. 通过分析模型的解的层次性特点, 提出基于自适应大规模邻域搜索框架的启发式算法. 通过多种规模对比实验发现, 所提出算法适用于大规模问题和即时性要求, 即在小规模算例下平均求解结果与精确解接近; 在中、大规模算例下平均求解结果相较于一般贪婪算法产生显著优化. 因此, 所提出模型和算法可为多结构型任务驱动的上门服务调度提供参考.

**关键词:** 上门服务; 拖期惩罚; 多结构型任务; 劳动力调度和路径规划问题; 自适应大规模邻域搜索

中图分类号: F270.7; TP18

文献标志码: A

DOI: 10.13195/j.kzyjc.2022.1281

**引用格式:** 展月, 姜兆勤, 刘振元. 多结构型任务驱动的上门服务调度优化模型与自适应大规模邻域搜索算法[J]. 控制与决策, 2024, 39(3): 947-955.

## Optimization model and revised adaptive large neighbourhood search algorithm for multi-structured on-site service scheduling problem

ZHAN Yue, JIANG Zhao-qin, LIU Zhen-yuan<sup>†</sup>

(School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China)

**Abstract:** This paper investigates a special type of on-site service scheduling problem that has the requirements of general on-site service scheduling problems, i.e., they require service workers, who may have different skill levels, to start from the same station, perform the assigned task in a path and return to the station. While the assigned task corresponds to a node within the graph in existing studies, realistic on-site service tasks may also have internal structure (known as multi-structured tasks), and thus the path generation process cannot be determined by the task sequence itself, and it is necessary to consider the co-optimization of task assignment and subgraph routing with selection of exit nodes and entry nodes. This paper analyses the characteristics of the problem and develops a mixed integer programming model with the objective of minimising the total tardiness. By analyzing the hierarchical characteristics of the solution, this paper proposes a heuristic algorithm based on the adaptive large neighbourhood search (ALNS) framework. Through various scale comparison experiments, it is found that the proposed algorithm is suitable for large scale problems and instantaneous requirements. The average solution result is close to the exact solution in small-scale cases; and in medium- and large-scale cases, the average solution result is significantly optimized compared to the general greedy algorithm. Therefore, the proposed model and algorithm can be used as a reference for multi-structured task-driven on-site service scheduling.

**Keywords:** on-site service; tardiness penalty; multi-structure task; workforce scheduling and routing problem; adaptive large neighbourhood search

收稿日期: 2022-07-18; 录用日期: 2022-11-22.

基金项目: 国家自然科学基金项目(72071087).

责任编辑: 刘士新.

<sup>†</sup>通讯作者. E-mail: zyliu@mail.hust.edu.cn.

\*本文附带电子附录文件, 可登录本刊官网该文“资源附件”区自行下载阅览.

## 0 引言

在现代服务业的各类场景,如IT运维服务、家庭医疗服务、家政服务,存在由服务人员直接转移至需求区域并开展服务的过程,称为上门服务(on-site service). 对上门服务的优化服务调度有助于提升服务整体的组织效率.

在相关文献中,该类调度问题对应劳动力调度和路径规划问题<sup>[1]</sup>,已有研究常在路径规划问题框架下将任务对应为图(graph)中的点,服务人员行进的路径即为图中的点的序列,点的序列可由任务的执行顺序直接确定. 现实上门服务过程中,如对街道或建筑物进行巡回式消毒杀菌时,每个任务往往还可能具备内部路径,即1个任务可能对应图中的多个点,需要在分配任务的同时确定每个任务的内部路径,并确定从哪个点进入该任务,从哪个点离开该任务并前往下一个任务. 此外,由现实服务过程归纳得到的要求,如任务具有拖期惩罚级别差异、服务人员可执行任务的差异、执行任务的时长差异,这些情形均对调度问题的模型建立和有效解决提出挑战. 带有任务与图中的点并非一一对应的特征这一问题,称为多结构型任务驱动的上门服务调度问题,其同时具有多技能人员调度问题<sup>[2]</sup>、劳动力调度和路径规划问题<sup>[1]</sup>的特性.

对于多技能人员调度问题,人力资源的多技能属性同时引入灵活度和困难度,已有许多学者对该问题展开了研究<sup>[2-5]</sup>. Afshar-Nadjafi<sup>[6]</sup>对截至2021年的160篇多技能人员调度问题的文献开展统计后得出,劳动力成本和运营时长是最常被考虑的两个目标,元启发式算法和精确算法是最常用的两类求解方法. 这些文献中,少有将任务的总拖期最短作为目标开展研究,类似的有Othman等<sup>[7]</sup>将等待时长和响应时长作为目标进行考虑.

劳动力调度和路径规划问题(workforce scheduling and routing problem, WSRP)由Castillo-Salazar等<sup>[1]</sup>于2016年提出. 相关文献基于电信维修、上门医疗服务、社区公共服务等不同问题场景<sup>[8-12]</sup>. 在问题目标上,除常见的将减少运营成本作为目标, Lu等<sup>[13]</sup>针对上门医疗服务提出了最小化器械损耗和最大化患者满意程度的目标;在问题定义上,相关文献研究了以技能要求、单个或多个交通方式、团队组建、路线安排、任务的时间窗、任务的优先级等为约束和/或决策变量的情况<sup>[1]</sup>;在解决方法上,基于局部搜索的元启发式方法广泛应用于该问题的求解<sup>[14]</sup>.

此外,在WSRP的扩展问题研究方面,Çakırgil等<sup>[15]</sup>提出了多技能人员调度和路径问题,运用两阶

段的数学启发式算法进行求解. 李珍萍等<sup>[16]</sup>提出了带时间窗和服务顺序约束的多需求车辆路径问题,考虑了多车和多种服务的情形,并使用联合优化遗传算法进行求解. Kovacs等<sup>[17]</sup>提出了服务技术人员路由和调度问题(service technician routing and scheduling problem, STRSP),考虑了服务外包和员工分组情形,并采用自适应大规模邻域搜索算法进行求解. Zamorano等<sup>[18]</sup>以电信维修为背景,分析了带有技术人员技能差异、技术人员分组、任务带有技能要求、任务带有跨天数时间窗的问题特点,并利用分支定界算法来求解. Mathlouthi等<sup>[19-21]</sup>提出了STRSP一般化的混合整数线性规划模型,利用分支定界算法精确求解了45个任务的算例,并设计了基于禁忌搜索的元启发式方法以求解含200个任务的大规模算例. Pillac等<sup>[22]</sup>提出了一种并行的自适应大规模邻域搜索算法,用来求解动态STRSP问题,并验证了该文献提到的op\_3算子表现优于其他算子. Si等<sup>[23]</sup>开发了一种面向交通的跨区域机会维护算法,用于解决多中心服务网络的运维优化和实时规划问题.

多技能人员调度问题、劳动力调度和路径规划问题均已被证明是NP-hard问题<sup>[1]</sup>. 二者因丰富的实际应用场景而拥有众多变体. 在对任务进行建模时,已有的研究文献侧重于分析任务的时间属性,如任务的时间窗、任务间的时间约束以及任务持续时长等. 对于任务的空间性,现有研究认为任务在空间上是无差别的,即任务的空间性结构被抽象为图上的点. 本文涉及的多结构型任务驱动的上门服务调度研究内容如下.

1) 考虑人员的多技能性,同时引入现实问题引致的任务的多结构型的特征,对任务的空间特征进行刻画;

2) 以总拖期最小为目标,建立针对多结构型任务驱动的上门服务调度的混合整数规划模型;

3) 设计基于自适应大规模邻域搜索(ALNS)的启发式算法,分别在任务环和地理环层面提出目标优化的自适应算子,并对接受机制进行改进,以保证算法在不同阶段的搜索性能;

4) 通过多组不同规模的算例开展计算实验,验证模型的正确性和算法的有效性.

## 1 问题描述与问题建模

### 1.1 问题描述

经典的上门服务调度问题给任务的执行者(即服务人员)安排任务、制定顺序,在以时间窗约束为例的若干约束下使得以总拖期为例的成本型目标函数

最小化(或使得收益型目标函数最大化).多结构型任务驱动的上门服务调度问题在此基础上,不仅为服务人员安排任务、制定顺序,还为每个服务人员在执行各任务时指定其内部路径.为刻画任务的内部路径,服务人员以一定顺序执行各任务的概念与服务人员以一定路径经过各点的概念相分离,分别称为任务点和地理点.

依据任务点与地理点的对应关系,可将任务点分为3类.其中:对应1个地理点的任务称为点型任务,进入该任务与离开该任务对应的地理点相同;包含2个地理点的任务称为线型任务,需要选择其中1个地理点作为进入该任务的地理点,任务完成后从另一个地理点离开;包含3个及以上的任务称为面型任务,在分配任务时需要同时指定进入该任务与离开该任务对应的地理点.特别地,本文考虑带有限制条件的面型任务,即仅包含3个对外地理点的面型任务.

在上述定义下,服务人员经过的任务点的序列称为任务环,服务人员经过的地理点的序列称为地理环.图1为多结构型任务的上门服务调度的任务环示例.其中:任务1~任务3、任务5、任务7~任务9为点型任务;任务4和任务6为线型任务,各有2个地理点;任务10为面型任务,有3个地理点.图1中,人员a将被指派依次执行任务8、任务1、任务2.因此,人员a对应的任务环为:任务开始→任务8→任务1→任务2→任务结束.

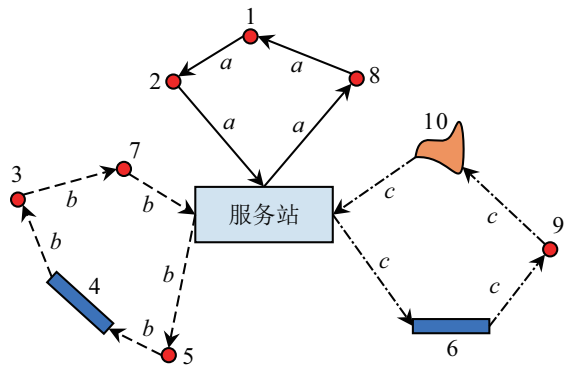


图1 多结构型任务驱动的上门服务调度的任务环示例

图2为多结构型任务驱动的上门服务调度中的地理环示例.其中:服务人员在执行任务6时,将从

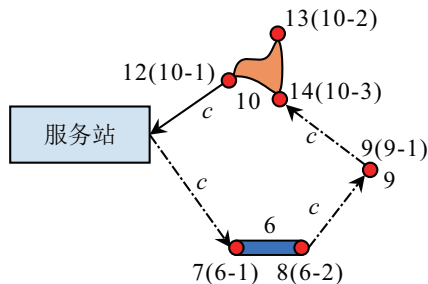


图2 多结构型任务驱动的上门服务调度的地理环示例

地理点7进入,从地理点8离开;在执行任务10时,将从地理点14进入,经过地理点13后从地理点12离开.因此,该服务人员对应的地理环为:服务站→地理点7→地理点8→地理点9→地理点14→(地理点13→)地理点12→服务站.

将地理点按其所属的任务点进行编号,则上述地理环也可表达为:服务站→地理点6-1→地理点6-2→地理点9-1→地理点10-3→(地理点10-2→)地理点10-1→服务站.此时,地理点的编号方式有总编号和“任务编号-内部编号”2种,如“地理点6-1”表示任务点6对应的1号地理点,即地理点7.在表达地理环时,不与其他任务点包含的地理点相连的地理点可在地理环的描述中省略.后文将采用省略形式的描述规则.

此外,多结构型任务驱动的上门服务调度问题考虑服务人员能够执行的任务差异,以及因服务人员能力差异导致的不同服务人员完成同一任务的时长差异.每个任务 $m$ 均对应已知的时间窗 $[R_m, P_m]$ .其中: $R_m$ 为任务 $m$ 的释放时刻,服务人员必须在此时刻后才能执行此任务,若服务人员提前到达该任务对应的任一地理点,则需先等待直至任务释放; $P_m$ 为任务 $m$ 的计划完成时刻,根据服务协议,在 $P_m$ 时刻后才完成任务 $m$ 视为延时完成,将产生拖期,否则拖期为0.现实情况下,由于服务人员的执行能力、交通资源等限制,服务人员可在 $R_m$ 时刻之前或之后到达相应地理点,任务 $m$ 也可在 $P_m$ 时刻后完成,因此,上述时间窗属于软时间窗.

### 1.2 模型建立

模型的目标为所有任务的总拖期最小,即

$$\min Z = \sum_{m \in C} (\hat{f}_m - P_m)^+ \quad (1)$$

模型的约束有

$$\sum_{s \in S} \sum_{n \in C \cup \{M+1\} \setminus \{m\}} x_{mn}^s = 1, \forall m \in C; \quad (2)$$

$$\sum_{s \in S} \sum_{n \in C \cup \{M+1\}} x_{0n}^s \leq S; \quad (3)$$

$$\sum_{m \in M_s^{(P)}} x_{0m}^s \leq 1, \forall s \in S; \quad (4)$$

$$\sum_{m \in M_s^{(P)}} x_{m, M+1}^s \leq 1, \forall s \in S; \quad (5)$$

$$\sum_{n \in C \cup \{M+1\}} x_{mn}^s - \sum_{n \in C \cup \{0\}} x_{nm}^s = 0, \quad \forall m \in M_s^{(P)}, s \in S; \quad (6)$$

$$\sum_{n \in C \cup \{M+1\} \setminus \{m\}} x_{mn}^s = y_{sm}, \forall m \in C, s \in S; \quad (7)$$

$$\sum_{i \in N_m} I_i = \sum_{i \in N_m} O_i = 1, \forall m \in C; \tag{8}$$

$$I_i + O_i = 1, \forall i \in N_m, m \in M^{(X)} \cup M^{(M)}; \tag{9}$$

$$\hat{t}_{mn} = \sum_{i \in N_m, j \in N_n} \tilde{T}_{ij} O_i I_j, \forall m \in C \cup \{0\},$$

$$n \in C \cup \{M+1\} \setminus \{m\}; \tag{10}$$

$$\hat{t}_{mn} \geq \sum_{i \in N_m, j \in N_n} \tilde{T}_{ij} (O_i + I_j - 1), \forall m \in C \cup \{0\},$$

$$n \in C \cup \{M+1\} \setminus \{m\}; \tag{11}$$

$$(\hat{f}_m + \hat{t}_{mn}) x_{mn}^s \leq \hat{s}_n, \forall m \in C \cup \{0\},$$

$$n \in C \cup \{M+1\} \setminus \{m\}, s \in S; \tag{12}$$

$$\hat{f}_m + \hat{t}_{mn} + (x_{mn}^s - 1) \Lambda \leq \hat{s}_n, \forall m \in C \cup \{0\},$$

$$n \in C \cup \{M+1\} \setminus \{m\}, s \in S; \tag{13}$$

$$\hat{f}_m = \hat{s}_m + \sum_{s \in S} y_{sm} Q_{sm}, \forall m \in C; \tag{14}$$

$$\hat{f}_m \leq T, \forall m \in C; \tag{15}$$

$$\hat{s}_m \geq R_m, \forall m \in C; \tag{16}$$

$$x_{mn}^s \in \{0, 1\}, \forall m, n \in M_s^{(P)}, s \in S; \tag{17}$$

$$y_{sm} \in \{0, 1\}, \forall m \in M_s^{(P)}, s \in S; \tag{18}$$

$$I_i, O_i \in \{0, 1\}, \forall i \in \tilde{N} \setminus \{0, N+1\}. \tag{19}$$

其中:式(2)表示所有任务均应被执行;式(3)和(4)表示服务人员均从服务站出发,到服务站结束;式(5)为任务的被执行关系;式(6)为网络流平衡约束;式(7)为任务执行与任务指派间的关系;式(8)表示每个实任务有且仅有一个进入点和一个离开点;式(9)表示面型任务和线型任务的每个地理点均不能同时作为进入点和离开点,即不能从相同的地理点开始和结束任务;式(10)为任务的进入点和离开点给定后任务间的里程时长换算;式(12)为任务环上相邻任务的时间约束;式(11)和(13)为对应的线性化约束,其含义与非线性约束相同,其中 $\Lambda$ 为一极大正整数;式(14)为任务的开始时刻与结束时刻的关系;式(15)表示所有任务均必须在时间周期内被处理;式(16)表示所有任务均必须在释放时刻后才能被处理;式(17)~(19)为决策变量的取值范围.

## 2 算法设计

对于多结构型任务驱动的上门服务调度问题,若不考虑任务的多结构型特性,则问题退化为劳动力调度和路径规划问题;若不考虑任务间的转移路径,则问题退化为多技能人员调度问题.由于劳动力调度、路径规划问题和多技能人员调度问题已被证明为NP-hard问题<sup>[1-2]</sup>,本文问题也为NP-hard问题,从而

无法在多项式时长内求得精确解.一般通用求解器如CBC、ABACUS、MINTO、CPLEX、Xpress、Gurobi等求解该问题的能力有限.综合考虑模型的目标函数可导性、决策变量定义域的规则性、由决策变量形成的解的邻域构造难易程度,并考虑为达成快速求解各种规模算例的目标,本文选择ALNS框架作为算法基础.

ALNS是Ropke等<sup>[24]</sup>针对带有时间窗的收送一体化车辆路径问题,在Shaw<sup>[25]</sup>的大邻域搜索算法的基础上提出的扩展算法<sup>[26]</sup>.作为一种启发式方法,ALNS可兼容多个根据问题本身特征而设计的破坏-修复算子,具有求解速度快、求解规模大、能够自我调整算子使用概率的特点.

ALNS算法首先利用贪婪算法获取问题的初始解,将其设置为初始全局最优解和局部最优解;然后利用若干破坏(destroy)算子和修复(repair)算子对当前局部最优解进行处理,即进行搜索过程.接受机制将对破坏-修复操作结束后得到的解进行取舍,并记录算子被使用的情况.经过一定搜索次数后,根据评分对每个算子进行权重更新.重复搜索操作,直至满足搜索过程的截止条件.

本节将依次介绍解的编码-解码方式、获取初始解的响应式贪婪算法、针对任务指派的算子和针对路径规划的算子、接受机制的设计、自适应层的设计和初始解修正机制.

### 2.1 解的两层编码

第1.2节提出的混合整数规划模型中问题的决策变量可描述为 $\mathcal{S} = (x_{mn}^s, y_{sm}, \hat{s}_m, \hat{f}_m, I_i, O_i, \hat{t}_{mn})$ ,由于该问题最终为每个服务人员确定其任务环和地理环,从而最优化问题的解可编码为 $ENC(\mathcal{S}) = (\mathcal{M}, \mathcal{R})$ .其中: $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_S\}$ 可视作位于任务层,表示各服务人员的任务点序列(即任务环中的实任务部分)的集合, $\mathcal{M}_s$ 为服务人员 $s$ 的任务

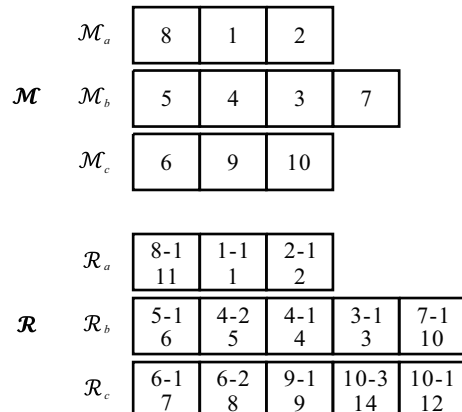


图3 解的2层编码

点序列,任务点的先后顺序为任务的执行顺序,若服务人员  $s$  不执行任何任务,则  $\mathcal{M}_s = \emptyset$ ;  $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_S\}$  可视为位于地理层,表示各服务人员的地理点序列(即地理环中对应实任务的地理点部分)的集合,  $\mathcal{R}_s$  为人员  $s$  的地理点序列,从而同理  $\mathcal{R}_s$  也可能为空. 图3为图1对应的作为最优化问题的解的编码. 其中:  $\mathcal{M} = \{\mathcal{M}_a, \mathcal{M}_b, \mathcal{M}_c\}$ ,  $\mathcal{R} = \{\mathcal{R}_a, \mathcal{R}_b, \mathcal{R}_c\}$ .

## 2.2 响应式贪婪算法获取初始解

本节参考Zhan等<sup>[27]</sup>给出的到达时刻、任务释放时刻、开始执行时刻间的关系,采用基于规则的贪婪算法获取算法的初始解. 具体步骤如下.

step 1: 将实任务集合  $\mathcal{C}$  中的所有任务  $m$  按释放时刻  $R_m$  升序排列,并初始化每个服务人员的空闲时刻  $K_s \leftarrow 0, \forall s \in \mathcal{S}$ .

step 2: 选取未被执行的任务中释放时刻最早的任务  $m$ , 根据  $\mathbf{M}_s^{(P)}$  找到所有可执行此任务的服务人员. 对于每个服务人员  $s \in \{s | m \in \mathbf{M}_s^{(P)}\}$ , 按  $\tilde{T}_{ij}$  和  $K_s$  求得该服务人员到达任务点  $m$  的最早时刻, 确定开始时刻. 将开始时刻加上执行时长  $Q_{sm}$  作为最早完成时刻. 将当前任务  $m$  加入对应人员的任务环中, 并将此服务人员的空闲时刻更新为任务  $m$  的完成时刻.

step 3: 判断是否完成所有任务的分配,若还有任务没有被分配,则执行 step 2; 否则,执行 step 4.

step 4: 所有任务已完成人员指派,开展对每个服务人员的路径优化. 由于任务点与地理点具备隶属关系,从而任务点序列与地理点序列也相互关联. 路径优化的具体步骤为,对于每个服务人员  $s$ :

step 4.1: 根据服务人员  $s$  的任务序列  $\mathcal{M}_s$ , 获取其中所有的线型任务(记其总任务数为  $n_s$ ) 和面型任务(记其总任务数为  $k_s$ ), 建立任务栈.

step 4.2: 建立深度为  $(n_s + k_s + 1)$ , 叶子节点数为  $n_s^2 k_s^6$  的决策树, 每个叶子节点对应一个地理点序列. 分别计算所有地理点序列的总拖期, 总拖期最短的地理点序列即为  $R_s$ .

## 2.3 自适应大规模邻域搜索算法

求解本问题涉及的ALNS算法主要包括启发式算子设计、接受机制设计以及自适应层改进. 本节对破坏算子和修复算子的设计主要参考了Kovacs等<sup>[17]</sup>、Shaw<sup>[25]</sup>、李珍萍等<sup>[28]</sup>、Graf<sup>[29]</sup>和Zhao等<sup>[30-32]</sup>的工作,其中具有较强随机性的算子将扩大算法运行时的搜索空间,具有较强择优性的算子将改进当前解,其中建立在地理环与任务环的插入位置上的优劣评价指标则需根据问题特征和解的编码特征设

计. 后文中算子对解的作用均以图3为例.

### 2.3.1 针对任务指派的破坏算子和修复算子

以下为3个针对任务指派的破坏算子.

1) 针对任务指派的随机破坏算子  $\text{rdm}$ .

随机选择一定数量的任务点(此处的一定数量依据问题算例规模改变,与总任务数有关),将其从当前任务点序列中移除,如随机移除  $\mathcal{M}_b$  内的任务5.

2) 针对任务指派的最差破坏算子  $\text{wdm}$ .

在当前的局部最优解中,将所有任务按拖期排序,优先移除拖期较长的任务点. 该算子能够尽量保留先前迭代过程的成果,任务拖期越小,任务被破坏的可能性越小. 若经计算当前解中拖期最长的任务点为5,则该算子将以最大概率移除  $\mathcal{M}_b$  内的任务5.

3) 针对任务指派的时间相关破坏算子  $\text{trdm}$ .

对于任意2个相异任务计算其时间相关度,优先移除时间相关度较高的2个任务点. 任意2个相异任务的时间相关度定义为

$$l_{mn} = |R_m - R_n| + |P_m - P_n|, \forall m \neq n \in \mathcal{C}. \quad (20)$$

当2个任务的时间相关度较高时,二者的时间窗重合度较高,从而在将此2个任务同时移除时交换成功得到新解的概率较高;当2个任务的时间相关度较低时,任务被移除后,因执行约束或时间约束,导致修复过程只能将被移除任务放回原有位置,进而导致无效破坏-修复活动的概率较高. 若经计算任务5与任务6的时间相关度最高,则同时移除  $\mathcal{M}_b$  内的任务5和  $\mathcal{M}_c$  内的任务6.

以下为3个针对任务指派的修复算子.

1) 针对任务指派的贪婪修复算子  $\text{grm}$ .

假定每个待插入任务均已选择最佳位置,计算插入后的总拖期增加值  $\Delta D$ . 按总拖期增加值将待插入任务升序排列,按排列顺序修复被破坏的任务. 插入点位选择该任务对应的最佳位置. 若经计算,待插入的任务5在插入  $\mathcal{M}_b$  内的任务4前导致的总拖期增加值最小,则在  $\mathcal{M}_b$  内的任务4前插入任务5.

2) 针对任务指派的次优修复算子  $\text{o2rm}$ .

次优修复算子同样需要给出任务修复顺序和任务插入点位. 首先随机选择待修复的任务,然后将该任务插入至使得总拖期增加第2小的点位上,即选择一个次优插入位. 该算子主要用于丰富解的选择空间. 如随机选择任务6,若经计算任务6在插入  $\mathcal{M}_c$  内的任务9前导致的总拖期增加值第2小,则在  $\mathcal{M}_c$  内的任务9之前插入任务6.

3) 针对任务指派的后悔修复算子  $\text{rrm}$ .

算子  $\text{rrm}$  基于算子  $\text{grm}$ , 即不仅需要计算最优插

入位置的预计拖期增量 $\Delta D_m^{(1)}$ ,还需要计算次优插入位置的预计拖期增量 $\Delta D_m^{(2)}$ ,并根据后者与前者之差计算任务 $m$ 对应的后悔值. 优先选择后悔值最大的任务进行修复,插入位置选择该任务对应的最佳位置. 若经计算,待插入的任务5在插入 $\mathcal{M}_b$ 内的任务4前导致的最优拖期增量与插入 $\mathcal{M}_c$ 内的任务9前导致的次优拖期增量得到的后悔值最大,则在 $\mathcal{M}_b$ 内的任务4前插入任务5.

### 2.3.2 针对路径规划的破坏算子和修复算子

为严格满足任务点与地理点间的隶属约束,针对路径规划进行破坏操作时需要同时对同一任务的任务点进行绑定,即若选择移除某个线型任务或面型任务的地理点,则需要同时对该任务点对应的其余地理点同时进行破坏. 由此,本节提出以下3个破坏算子.

#### 1) 针对路径规划的随机破坏算子 rdr.

随机选择一定数量任务点将其移除,同时保证线型任务和面型任务的任务点被选择的概率高于点型任务的任务点,如以较大概率移除 $\mathcal{M}_c$ 内的面型任务10. 由于选择线型任务和面型任务的出入点对地理点路径的长度改变影响较大,进而对总拖期的改变影响较大,有必要保证对足够多的线型任务和面型任务的各种出入点可能性开展搜索.

#### 2) 针对路径规划的最差破坏算子 wdr.

计算每个任务点被移除后与被移除前的目标函数差值(即总拖期). 差值较大的任务点优先被移除. 若经计算,任务5在移除后导致的总拖期减少量最大,则优先移除 $\mathcal{M}_b$ 内的任务5.

#### 3) 针对路径规划的位置相关破坏算子 lrdr.

lrdr与trdm思想类似,即同时移除位置相关度较高的任务点,然后在修复操作中以更大概率地将其指派给同一个服务人员来达到位置聚集的效果,从而减少转移时长. 本节定义的位置相关度为2个地理点间的欧氏距离的倒数. 若经计算,任务5与任务6的位置相关度最高(即欧氏距离最小),则同时移除 $\mathcal{M}_b$ 内的任务5和 $\mathcal{M}_c$ 内的任务6.

以下为2个针对路径规划的修复算子.

#### 1) 针对路径规划的就近修复算子 nrr.

定义每个地理点序列的中心为所有地理点坐标的均值 $(\bar{X}, \bar{Y})$ ,则可定义聚集度 $C_{n\mathcal{R}_s}$ 为任务点 $n$ 插入地理点序列 $\mathcal{R}_s$ 后,此任务点到地理点序列中心的欧氏距离. 该算子优先修复聚集度较小的地理点对应的任务点. 若任务点属于面型任务或线型任务,则需要计算任务的中心到地理点序列中心的欧氏距离. 修复位置选择最佳位置. 若经计算,待插入的任

务10的中心点到 $\mathcal{R}_c$ 的中心点的欧氏距离最小,则优先修复任务10,且若经计算,待插入的任务10在插入 $\mathcal{M}_c$ 内的任务9后的总拖期最小,则将任务10插入 $\mathcal{M}_c$ 内的任务9后.

#### 2) 针对路径规划的贪婪修复算子 grr.

grr类似于grm,即将任务点按修复前后的目标函数差值升序排列,由此确定任务点恢复的顺序. 若经计算,待插入的地理点5-1在插入 $\mathcal{R}_b$ 内的地理点4-2前导致的总拖期增加值最小,则在 $\mathcal{M}_b$ 内的任务4前插入任务5.

### 2.3.3 基于模拟退火算法的接受机制

将一定迭代次数确定为一个周期 $\sigma$ ,总周期数记为 $\Sigma$ . 当新解不导致目标函数的优化时,接受新解的概率为

$$P_{\text{accept}}(\Delta Z) = \exp\left(-\frac{\Delta Z}{T_\sigma}\right). \quad (21)$$

其中: $\Delta Z$ 为目标函数增量; $T_\sigma$ 为当前迭代周期的温度,在迭代周期结束后发生改变, $T_\sigma$ 的改变机制即为降温策略. 升温的概率<sup>[25]</sup>为

$$P_{T_{\sigma+1}}^{\text{up}}(Z_\sigma^{(f)}, Z_\sigma^{(s)}, \eta) = \exp\left(\frac{Z_\sigma^{(f)} - Z_\sigma^{(s)}}{\Sigma - \sigma}\right). \quad (22)$$

其中: $Z_\sigma^{(f)}$ 、 $Z_\sigma^{(s)}$ 分别为第 $\sigma$ 周期开始和结束时的目标函数值, $(\Sigma - \sigma)$ 为剩余迭代次数. 以上接受机制用于保证不同阶段的搜索效果,具体表达如下.

1) 迭代初期, $T_\sigma$ 较低,目标函数下降较快,不导致目标函数产生优化的解将有较大概率被舍弃,从而迭代初期解可快速收敛;

2) 迭代中期,目标函数变化量减小,将可能遇到陷入局部最优情况,而此时 $T_\sigma$ 上升的概率变大,从而以更大概率地接受较差解,将有利于跳出局部最优;

3) 迭代后期,剩余迭代次数 $(\Sigma - \sigma)$ 减小, $T_\sigma$ 上升概率减小,仅接受改进解趋于收敛,将避免迭代后期结果变差.

### 2.3.4 自适应层的设计

每次迭代,经过算子组合计算得到的新解有3种情况,即新解优化了目标函数并成为新的局部最优、新解没有优化但是被接受、新解没有优化且被舍弃. 一次迭代完成后可对算子进行评分,上述的3种情况分别对应不同的分数. 每个迭代周期结束后,可对不同决策的权重进行一次更新. 记算子索引 $d \in \{1, 2, \dots, D\} = \mathbf{D}$ ,算子 $d$ 的累计使用次数为 $u_d$ (每用到1次, $u_d$ 增加1),算子 $d$ 的累计评分为 $\pi_d$ (每用到1次,上述3种情况使得 $\pi_d$ 对应地增加 $\tau_1$ 、 $\tau_2$ 、 $\tau_3$ ,且 $\tau_1 > \tau_2 > \tau_3$ ),折现系数记为 $\rho$ ,则对于全体算子 $d$ ,其权

重  $w_d$  的更新公式为

$$w_d \leftarrow \begin{cases} (1 - \rho)w_d + \rho \cdot 0, & u_d < 1; \\ (1 - \rho)w_d + \rho \frac{\pi_d}{u_d}, & u_d \geq 1. \end{cases} \quad (23)$$

算子的调用概率起初均等,更新后的算子调用概率为各算子权重的归一化结果,从而权重大的算子更有可能被选到.自适应层可根据不同的算例和不同的搜索过程自行改变各算子的使用概率.

### 3 计算实验

本节利用多组基于实际应用背景整理生成的小规模、中规模、大规模算例进行求解实验,并通过 CPLEX 求解器的求解结果进行对比,以验证混合整数规划模型的正确性和 ALNS 算法的有效性.

算法在 Windows 10 下利用 JDK 14.0 使用 Java 语言编程实现.所有计算实验在 Intel Core i5-10400 CPU 2.90 GHz, 8 GB RAM 上运行. CPLEX 版本为 12.6.3.

#### 3.1 算例说明

算例数据来源于 IT 上门服务调度背景,参考相关文献的算例表示方法自主生成算例.每个算例包含算例基本信息(算例名、任务数量、员工数量以及服务站坐标)、任务列表和执行人员列表.点、线、面型任务数量的比值为 4 : 3 : 3.且保证:

1) 每个任务至少对应 1 名可完成该任务的服务人员.

2) 任务的释放时刻是随机生成的,每个类型的任务对应的完成时刻的范围不同.

3) 使用 2 元组  $(S_{\text{num}}, M_{\text{num}})$  表示一组算例.其中:  $S_{\text{num}}$  为服务人员数量,  $M_{\text{num}}$  为任务点数量.一组算例中的 1 个算例  $i$  记为  $(S_{\text{num}}, M_{\text{num}})_i$ .

计算实验测试的算例分为小、中、大 3 个规模.小规模服务人员数量  $S_{\text{num}} \in \{3, 5\}$ ,任务点数量  $M_{\text{num}} \in \{10, 14, 17\}$ ;中规模服务人员数量  $S_{\text{num}} \in \{5, 10, 15\}$ ,任务点数量  $M_{\text{num}} \in \{20, 40, 50, 60\}$ ;大规模服务人员数量  $S_{\text{num}} \in \{20, 25\}$ ,任务点数量  $M_{\text{num}} \in \{80, 100, 120\}$ .同时保证任务点数量与服务人员数量的比值不超过 10.每组小规模算例各包含 10 个算例,每组中规模 and 大规模算例各包含 5 个算例.

#### 3.2 参数设置

预实验部分要求确定每周期迭代次数、总迭代次数和评分  $(\tau_1, \tau_2, \tau_3)$  的具体值,为此以 ALNS 算法结果的目标函数值为评价指标开展正交实验.其中,综合考虑相关论文算法设定、算法执行速度等因素,考虑总迭代次数在  $\{2000, 5000, 10000, 15000\}$  中选

择,每周期迭代次数在  $\{50, 100, 150, 200\}$  中选择.实验结果显示:目标函数值随总迭代次数增加而减小,而当总迭代次数大于 10000 后,对于其他各种条件而言增大总迭代次数均使得算法结果不再有明显提升,且算法运行时长随之线性增长,从而将总迭代次数确定为 10000 次.正交实验显示:  $(\tau_1, \tau_2, \tau_3)$  取  $(20, 3, 1)$  时结果最优且最稳定,  $(20, 5, 1)$  次之,  $(20, 10, 1)$  最差.从而将  $(\tau_1, \tau_2, \tau_3)$  确定为  $(20, 3, 1)$ .每周期迭代次数取 50 次时目标函数值最优,从而取 50 次迭代为一个周期.

#### 3.3 实验结果对比分析

小规模算例共有 60 个.首先利用 CPLEX 对每个小规模算例尝试求解精确解,限时为 3600 s;然后使用 ALNS 算法进行求解,每个算例进行 10 次实验.其中:使用 ALNS 求解每个算例后,应输出每个任务点对应的执行人员、离开点、进入点、开始时刻  $\tilde{s}$ 、结束时刻  $\tilde{f}$  和拖期  $\tilde{d} = (\tilde{f} - P)^+$ .

针对小规模算例的计算实验着重对比 CPLEX 与 ALNS 算法的求解效果.对于各自求解的结果应罗列每个算例的编号、ALNS 算法 10 次求解的平均时长和最优结果、CPLEX 求解的最优结果和求解时长、ALNS 算法最优解与 CPLEX 最优解的偏差以及 ALNS 算法平均解与 CPLEX 最优解的偏差,并着重呈现 CPLEX 求得全局最优解、ALNS 求得全局最优解以及在 CPLEX 未求得全局最优解时 ALNS 算法表现更优的情况.

CPLEX 可求解  $(3, 10)$ 、 $(5, 10)$ 、 $(5, 14)$  的全部算例、90% 的  $(3, 14)$  和 60% 的  $(5, 17)$  算例,且在求解  $(3, 10)$ 、 $(3, 14)$  的部分算例时求解时长过长.而 ALNS 不仅一次运行时的平均耗时在 1 s 内,求解结果与最优解平均相差也在 7% 以内.特别地,对于算例集  $(3, 17)$  而言,CPLEX 无法在 3600 s 内求得 9 个算例的精确解,而 ALNS 算法在运行相同时长后的结果均比 CPLEX 更优.

对于中规模 40 个算例和大规模 20 个算例,CPLEX 无法在 3600 s 内求得任一精确解.因此,考虑使用 ALNS 算法对每个中规模算例求解 10 次,对每个大规模算例求解 5 次. ALNS 算法将贪婪算法求得的解作为初始解,从而可计算 ALNS 算法求解结果的总拖期对贪婪算法求解结果的总拖期的偏差,并以此来衡量 ALNS 算法每次求解结果的优化率.多次求解中,总拖期最小的解称为该算例的近似最优解.

对于中、大规模算例集而言,ALNS 算法与贪婪算法的求解效果比较应罗列每个算例的规模、ALNS

算法的平均优化率、ALNS算法的平均求解时长以及获得近似最优解的次数与求解总次数之比。由计算实验可知,ALNS算法的平均优化率集中于20%~30%间。前6组算例中,任务数相同时,服务人员越少,平均求解时长越长;后6组算例中,任务数相同时,服务人员越少,平均求解时长越短。

此外,在ALNS算法的计算过程中可记录前文涉及的全体算子在迭代计算中的使用情况。依据记录结果,每个算子的表现和使用概率在迭代初期存在较大差异,但是随着迭代进行,特别是进入后期逐渐使得所得解收敛后,各算子被选择的概率具有较强的初值无关性。平均而言,算子o2rm、rrm和nrr的使用次数较多。

## 4 结论

本文研究了一类受现实上门服务问题启发的少见于已有文献的上门服务调度问题,其特征在于服务人员执行的任务具有多结构性,这要求任务点与地理点的概念分离,并要求在求解调度优化问题时同时为每个服务人员确定任务点序列和地理点序列。为此,本文通过问题特征分析建立了混合整数规划模型,并设计了用于求解该问题的贪婪算法和ALNS算法。通过多规模计算实验可确认,ALNS算法的主要优势为适用于即时性要求和大规模问题;对于小规模算例,ALNS算法的平均结果与CPLEX精确解相差在7%以内;对于中、大规模算例,相较于贪婪算法获得的初始解,ALNS算法使得总拖期下降了20%~30%。

ALNS算法在部分算例,如 $(5, 10)_3$ 与 $(5, 10)_{10}$ 算例中表现较差,这反映出ALNS算法在初值的进一步去依赖性和邻域设计上还有改进空间。此外,除了总拖期最小化这一目标以外,上门服务调度问题还应考虑减少路径成本、减少劳动力成本、满足个人对任务的偏好等目标。也应在多目标的模型框架下讨论如何提高上门服务的综合服务水平。从求解算法角度看,分析本文问题对应的线性化前的非线性模型,可能对开展精确求解算法的设计有启发意义。

## 参考文献(References)

- [1] Castillo-Salazar J A, Landa-Silva D, Qu R. Workforce scheduling and routing problems: Literature survey and computational study[J]. *Annals of Operations Research*, 2016, 239(1): 39-67.
- [2] 刘振元. 服务系统中的多技能人员调度[M]. 北京: 清华大学出版社, 2019: 5-11.  
(Liu Z Y. Multi-skilled workforce scheduling in service system[M]. Beijing: Tsinghua University Press, 2019: 5-11.)
- [3] Dupin N, Parize R, Talbi E G. Matheuristics and column generation for a basic technician routing problem[J]. *Algorithms*, 2021, 14(11): 313.
- [4] Frits M, Bertok B. Routing and scheduling field service operation by P-graph[J]. *Computers & Operations Research*, 2021, 136: 105472.
- [5] Demirbilek M, Branke J, Strauss A K. Home healthcare routing and scheduling of multiple nurses in a dynamic environment[J]. *Flexible Services and Manufacturing Journal*, 2021, 33(1): 253-280.
- [6] Afshar-Nadjafi B. Multi-skilling in scheduling problems: A review on models, methods and applications[J]. *Computers & Industrial Engineering*, 2021, 151: 107004.
- [7] Othman S B, Hammadi S, Quilliot A. Multi-objective evolutionary for multi-skill health care tasks scheduling[J]. *IFAC-PapersOnLine*, 2015, 48(3): 704-709.
- [8] Paraskevopoulos D C, Laporte G, Repoussis P P, et al. Resource constrained routing and scheduling: Review and research prospects[J]. *European Journal of Operational Research*, 2017, 263(3): 737-754.
- [9] Pourjavad E, Almeddawe E. Optimization of the technician routing and scheduling problem for a telecommunication industry[J]. *Annals of Operations Research*, 2022, 315(1): 371-395.
- [10] Pekel E. A simple solution to technician routing and scheduling problem using improved genetic algorithm[J]. *Soft Computing*, 2022, 26(14): 6739-6748.
- [11] Liu R, Yuan B, Jiang Z B. A branch-and-price algorithm for the home-caregiver scheduling and routing problem with stochastic travel and service times[J]. *Flexible Services and Manufacturing Journal*, 2019, 31(4): 989-1011.
- [12] Li Y, Ye C M, Wang H P, et al. A discrete multi-objective grey wolf optimizer for the home health care routing and scheduling problem with priorities and uncertainty[J]. *Computers & Industrial Engineering*, 2022, 169: 108256.
- [13] Lu M L, Qin Z, Cao Y M, et al. Scalable news recommendation using multi-dimensional similarity and Jaccard-Kmeans clustering[J]. *Journal of Systems and Software*, 2014, 95: 242-251.
- [14] Punyakum V, Sethanan K, Nitisiri K, et al. Hybrid differential evolution and particle swarm optimization for multi-visit and multi-period workforce scheduling and routing problems[J]. *Computers and Electronics in Agriculture*, 2022, 197: 106929.
- [15] Çakırgil S, Yücel E, Kuyzu G. An integrated solution approach for multi-objective, multi-skill workforce

- scheduling and routing problems[J]. *Computers & Operations Research*, 2020, 118: 104908.
- [16] 李珍萍, 张煜炜. 带时间窗和服务顺序约束的多需求车辆路径问题[J]. *控制与决策*, 2019, 34(7): 1565-1570.  
(Li Z P, Zhang Y W. Multiple demands vehicle routing problem with time windows and service order constraints[J]. *Control and Decision*, 2019, 34(7): 1565-1570.)
- [17] Kovacs A A, Parragh S N, Doerner K F, et al. Adaptive large neighborhood search for service technician routing and scheduling problems[J]. *Journal of Scheduling*, 2012, 15(5): 579-600.
- [18] Zamorano E, Stolletz R. Branch-and-price approaches for the multiperiod technician routing and scheduling problem[J]. *European Journal of Operational Research*, 2017, 257(1): 55-68.
- [19] Mathlouthi I, Gendreau M, Potvin J Y. Mixed integer linear programming for a multi-attribute technician routing and scheduling problem[J]. *INFOR: Information Systems and Operational Research*, 2018, 56(1): 33-49.
- [20] Mathlouthi I, Gendreau M, Potvin J Y. Branch-and-price for a multi-attribute technician routing and scheduling problem[J]. *Operations Research Forum*, 2021, 2(1): 1.
- [21] Mathlouthi I, Gendreau M, Potvin J Y. A metaheuristic based on tabu search for solving a technician routing and scheduling problem[J]. *Computers & Operations Research*, 2021, 125: 105079.
- [22] Pillac V, Guéret C, Medaglia A L. A fast reoptimization approach for the dynamic technician routing and scheduling problem[M]. *Cham: Recent Developments in Metaheuristics*, 2017: 347-367.
- [23] Si G J, Xia T B, Zhang K G, et al. Technician collaboration and routing optimization in global maintenance scheduling for multi-center service networks[J]. *IEEE Transactions on Automation Science and Engineering*, 2022, 19(3): 1542-1554.
- [24] Ropke S, Pisinger D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows[J]. *Transportation Science*, 2006, 40(4): 455-472.
- [25] Shaw P. Using constraint programming and local search methods to solve vehicle routing problems[C]. *International Conference on Principles and Practice of Constraint Programming*. Heidelberg, 1998: 417-431.
- [26] 杨艳艳. 基于ALNS算法的自动驾驶汽车配送路径规划[D]. 大连: 大连理工大学, 2020.  
(Yang Y Y. The self-driving car distribution route planning based on ALNS[D]. Dalian: Dalian University of Technology, 2020.)
- [27] Zhan Y, Wang Z Z, Wan G H. Home service routing and appointment scheduling with stochastic service times[J]. *European Journal of Operational Research*, 2021, 288(1): 98-110.
- [28] 李珍萍, 周文峰, 张煜炜, 等. 考虑卸载顺序约束的成品油二次配送车辆路径问题[J]. *控制与决策*, 2020, 35(12): 2999-3005.  
(Li Z P, Zhou W F, Zhang Y W, et al. Vehicle routing problem of refined oil secondary distribution considering unloading sequence constraints[J]. *Control and Decision*, 2020, 35(12): 2999-3005.)
- [29] Graf B. Adaptive large variable neighborhood search for a multiperiod vehicle and technician routing problem[J]. *Networks*, 2020, 76(2): 256-272.
- [30] Zhao Z Y, Liu S X, Zhou M C, et al. Heuristic scheduling of batch production processes based on petri nets and iterated greedy algorithms[J]. *IEEE Transactions on Automation Science and Engineering*, 2022, 19(1): 251-261.
- [31] Zhao Z Y, Liu S X, Zhou M C, et al. Dual-objective mixed integer linear program and memetic algorithm for an industrial group scheduling problem[J]. *IEEE/CAA Journal of Automatica Sinica*, 2020, 8(6): 1199-1209.
- [32] Zhao Z Y, Zhou M C, Liu S X. Iterated greedy algorithms for flow-shop scheduling problems: A tutorial[J]. *IEEE Transactions on Automation Science and Engineering*, 2022, 19(3): 1941-1959.

### 作者简介

展月(1997—), 女, 硕士生, 从事IT服务人员调度优化的研究, E-mail: 1603664625@qq.com;

姜兆勤(1999—), 男, 硕士生, 从事IT服务、施工服务、物流服务等研究, E-mail: kushim\_jiang@outlook.com;

刘振元(1973—), 男, 教授, 博士, 从事服务智能、生产计划与调度等研究, E-mail: zyliu@mail.hust.edu.cn.