

需求响应公交多车型混合运营调度

胡满江^{1,2}, 王兹珑¹, 王晓伟^{1,2†}, 刘雨诺¹, 曹恺^{1,3}, 边有钢^{1,2}

(1. 湖南大学 机械与运载工程学院, 湖南 长沙 410082;
2. 湖南大学 无锡智能控制研究院, 江苏 无锡 214123;
3. 东风悦享科技有限公司, 湖北 武汉 430058)

摘要:需求响应公交多采用包车、拼车模式运营,然而,面对时变道路信息,单一运营模式将影响运营效率。为最大化利用车辆资源,提出一种在有限公交资源和多种运营模式下的灵活调度与车队配置方法。首先,基于包车、拼车模式特点,结合新型点对点拼车模式和异质车队特性,统筹考虑包车、拼车、点对点拼车三种运营模式,通过设计双层时间窗,以最大化运营效益为目标,构建多车型混合运营调度模型。其次,改进自适应大邻域搜索的启发式算法,通过优化邻域搜索操作,提高求解质量。最后,基于武汉市真实路网数据和居民出行数据进行仿真试验。结果表明,本文提出的多车型混合运营模型对比包车、拼车模型,在车队规模达到100辆时,运营收益分别提升9%、7%,订单响应率提升10%、6%,乘客平均等待时间下降30%、25%,综合提升了运营商和乘客的效益。

关键词:需求响应公交;多车型;混合运营模型;点对点拼车;启发式算法

中图分类号:U491.1 文献标志码:A

Scheduling for Multi-vehicle-type Mixed Operation of Demand Responsive Transit

HU Manjiang^{1,2}, WANG Zilong¹, WANG Xiaowei^{1,2†}, LIU Yunuo¹, CAO Kai^{1,3}, BIAN Yougang^{1,2}

(1. College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China;
2. Wuxi Intelligent Control Research Institute, Hunan University, Wuxi 214123, China;
3. Dongfeng USharing Technology Co., Ltd., Wuhan 430058, China)

Abstract: Demand responsive transit typically operates in chartered or ride-sharing modes. Due to the time-varying road information, using a single operation mode will reduce the operational efficiency. To maximize the use of vehicle resources, a flexible scheduling and fleet configuration method under limited bus resources and multiple operation modes is proposed. First, based on the characteristics of chartered and ride-sharing modes, combined with the new point-to-point ride-sharing mode and heterogeneous fleet characteristics, the above three modes are taken into account as a whole. By designing a double-layer time window and aiming to maximize the operational benefits, a multi-vehicle-type mixed operation scheduling model is constructed. Secondly, the heuristic algorithm

* 收稿日期:2024-12-13

基金项目:国家自然科学基金资助项目(52172384), National Natural Science Foundation of China(52172384);湖南省科技创新计划项目(2022RC1033), Science and Technology Innovation Program of Hunan Province(2022RC1033)

作者简介:胡满江(1988—),男,山东临沂人,湖南大学教授,博士

† 通信联系人,E-mail:wxw9@163.com

of adaptive large neighborhood search algorithm is improved. By optimizing neighborhood search operations, the solution quality is enhanced. Finally, simulation experiments based on real road network data and travel data from Wuhan City are conducted. The results show that, compared with the chartered and ride-sharing models, when the fleet size reaches 100 vehicles, the multi-vehicle-type mixed operation model increases the operational revenue by 9% and 7%, improves the order response rates by 10% and 6%, and reduces the average passenger wait time by 30% and 25%, respectively. Overall, the model enhances both the operator's and passengers' benefits.

Key words: demand responsive transit; multi-vehicle-type; mixed operation model; point-to-point ride-sharing; heuristic algorithm

随着公共出行方式的日益多样化,传统公交服务站点固定、发车时刻僵化等局限性愈发凸显,越来越难以适应人们多样化、个性化的出行需求。在此背景下,需求响应公交逐渐成为公交发展的重要趋势。公交服务平台通过收集乘客的出行信息,设计车辆运营路线,有效提高公交服务效率及运营灵活性。此外,智能网联技术的发展为需求响应公交提供了技术支撑,通过搭载传感器和车联网设备,实现了人、车、路、云之间的互联互通,从而提供了更为高效和多样的响应式出行服务。同时,公交车辆资源日益丰富,车型发展呈现小型化、多样化趋势。在日常运营中,需求响应公交可采用包车模式,实现定制化出行服务,乘客通过移动端提交出行信息,平台在收集信息后进行车辆匹配。在包车模式的基础上,为进一步提高车辆利用率和需求响应效率,平台采用拼车模式,在收集车辆状态和乘客出行需求后,通过算法将起点、终点相近的乘客进行匹配,并分配至同一辆车进行接送。在此背景下,研究在有限车辆资源限制下的多车型混合运营调度方法,具有重要的理论和现实意义。

现有研究中,包车模式常考虑贪婪策略,通过预先设置目标,以当前状态的最优选择进行匹配。然而在现实情况中,车辆信息、乘客请求均为时变特性,因此包车模式的调度算法需进一步考虑动态性。Xu 等^[1]针对离线、在线场景,设计基于最小成本的网络流算法。Ding 等^[2]开发分层调度框架,将大规模道路网络划分为多个子区域,子区域内建立车乘关系的可更新调度策略。Gao 等^[3]通过预测方法制定引导策略,并根据引导策略计算车乘之间的最佳二分匹配。Van Engelen 等^[4]提出一种基于需求预测的在线动态插入算法,以解决新乘客需求产生时的订

单分配、插入和路径再规划问题。Li 等^[5]先利用二分图最大权重匹配方法进行需求匹配,然后通过预测未来各区域乘车需求,实现空车再分配。Meskar 等^[6]综合考虑司机拒乘、网络时空特征、车队规模动态变化等因素,为调度平台提供一种定价与匹配策略。Chen 等^[7]通过优化蒙特卡罗树搜索方法,解决多周期顺序调度问题,以此提升车乘匹配效率、降低订单取消概率,同时减少请求等待时间。You 等^[8]设计一种动态区域划分方法,用于识别实时需求的相应区域类型,同时结合区域多方补贴机制,为不同类型区域开发新需求分配策略,并引入一种多方补贴机制的参数优化方法。Guo 等^[9]结合出行距离、平台收入、整体供需平衡、乘客等待时间等多个目标,设计可解释整合模型,生成 Pareto 解集权衡各目标。Shi 等^[10]定义一个车辆价值函数,描述每个区域内车辆的时空价值,实现高效车乘匹配和闲置车辆调度。

相比于包车模式,拼车模式存在共享线路,且在线路执行期间,决策信息可能随时间改变,为了提高拼车效益,需实时改变规划结果。Huang 等^[11]采用基于区域的出行时间估计方法,将乘客位置封装至区域内,计算每个符合条件的乘客组合最短路径,选取最小额外行驶时间组合。Meshkani 等^[12]提出一种基于图的多对一拼车匹配算法,首先进行单对单匹配,随后解决最大权重匹配问题,以合并出行请求。Tuncel 等^[13]设计基于图的方法,通过定义新调度成本项,量化车辆调度对区域期望供给平衡的贡献值。Li 等^[14]考虑一对三匹配、订单实时插入等情况,将问题转化为多目标整数规划,并利用分层方法求解。Simonetto 等^[15]针对车队车辆与乘客请求之间的线性分配问题,提出一种基于联邦框架的高效动态拼车算法。Guo 等^[16]构建考虑动态时间和预期迁移

的拼车框架,采用多策略、图搜索的启发式算法,保
证求解质量.Ghandeharioun 等^[17]开发模块化实时模
拟框架,将拼车问题转换为具有动态性、确定性的
按需分配问题,降低算法的求解复杂度.Hyland
等^[18]研究共享乘车的优势,采用基于代理的随机动
态仿真框架,实现动态分配及路径规划策略.Yao
等^[19]以最小化车辆行驶距离为目标,提出一种动态
树算法,同时设计预处理方法处理待分配订单,进
一步提升算法性能.

综上所述,现有研究集中于包车、拼车模式的
车辆调度问题,在问题建模和算法求解方面成果丰
富.然而,面对时变的道路状况以及特殊出行时段
(如出行高峰期、客流聚集时段),单一的运营模式
会导致运行成本较高、订单响应率低、车辆利用率低
等问题,缺乏对需求响应公交在有限车辆资源和多
种运营模式下灵活调度与车队配置问题的深入考
虑.针对上述不足,本文基于包车、拼车模式特点,
提出一种点对点拼车模式.为了进一步提升乘客出
行体验和平台运营收益,设计双层时间窗,并以最
大化运营效益为目标,结合需求响应公交的异质车
队,以及三种运营模式,提出一种在有限车辆资源
限制下的多车型混合运营调度方法.之后,设计基

于自适应大邻域搜索(adaptive large neighborhood
search, ALNS)的求解算法,改进邻域搜索操作.最
后基于武汉实际道路信息,验证算法和模型性能,
并对模型关键参数进行敏感性分析.

本文研究包括如下内容:第 1 节阐述研究问题
并建立模型;第 2 节为模型设计求解算法;第 3 节基
于实际道路信息,对本文模型及求解算法进行验证
和参数敏感性分析;第 4 节进行总结并展望.

1 模型建立

本文以需求响应公交为研究对象,综合考虑运
营区域、乘客需求、异质车队相关要素,提出一种点
对点拼车模式,并构建多车型混合运营模型.

1.1 问题描述

在多车型混合运营(multi-vehicle-type mixed
operation, MVTMO)调度问题中,平台在运营区域
内集中管理需求响应公交车队,在运营时段内,平台
分配待处理订单,车辆根据分配结果运行,待运营
时段结束时停至停车点.如图 1 所示, o 、 d 表示订单
起始点、目的点,经匹配后,车辆 v_1 先后处理包车订
单, v_2 处理拼车订单, v_3 处理点对点拼车订单,虚线
为空载行驶,实线为载客行驶.

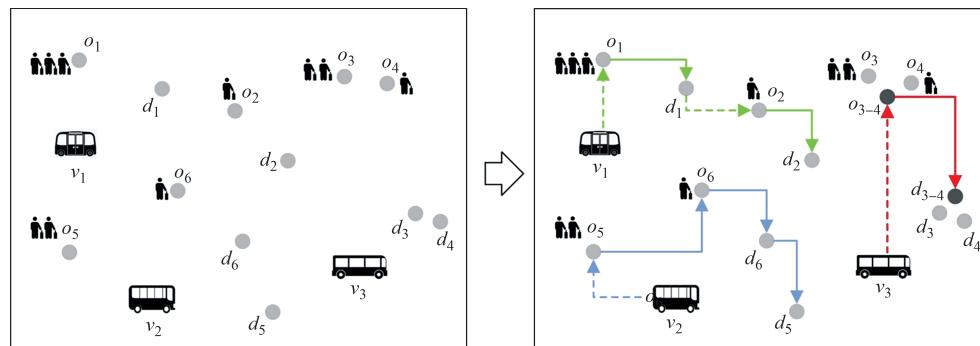


图 1 MVTMO 示意图

Fig.1 An illustration of MVTMO

因此,MVTMO 调度涉及的主要问题是基于异
质车队和个性化需求信息,合理进行车乘匹配,优
化车辆的服务路径,具体涉及以下基本要素:

1) 运营区域:将运营区域用可通行路网表示,
即加权有向图,由道路节点集合和有向道路集合构
成,有向道路权重代表道路的行驶距离.乘客可根
据自身需求选择合适的上、下车节点.

2) 个性化出行需求:在运营时段内,乘客根据
自身需求选择订单类型,传至服务平台,上传订单

信息包括订单类型 q 、乘车人数 n 、期望出发时间 t 、起
始点 o 、目的点 d 等.

3) 异质车队:考虑订单类型特点,本文采用异
质车队,不同车型在运营成本和额定载客量等方面
存在差异.在任一调度时刻,平台已知当前所有车
辆位置、剩余载客量、车速、当前服务订单和已绑定
订单信息,为待分配订单匹配车辆.

将运营时段均匀离散为 ρ 个时刻,即 $\{t_1, t_2, \dots, t_i, t_{i+1}, \dots, t_\rho\}$, 调度平台在每个调度时刻 t_i 进行决策,

制定短周期 t_i 和长周期 t_T , 平台为 t_T 内出发订单匹配车辆, 即在 $[t_{i-1}, t_i]$ 时段内的实时订单和 $[t_i, t_i + t_T]$ 内的预定订单. 对于 t_i 时刻匹配成功订单, 以短周期 t_i 为界, 出发时间在 $[t_{i-1}, t_i + t_i]$ 内为绑定订单, 不能被重新分配, 在 $[t_i + t_i, t_i + t_T]$ 内为灵活订单, 根据实时情况进行再分配. 因此, 对于下一调度时刻 t_{i+1} , 平台将完成新产生请求和灵活订单的再分配, 车辆行驶路线随之更新.

针对上文所研究的 MVTMO 调度问题, 为了清晰描述、构建模型, 做出如下假设:

- 1) 每个订单只能被服务一次, 匹配成功后, 乘客在期望出发时间前到达起始点, 不考虑临时取消行程或乘客数量临时变更等情况.
- 2) 假设所有车辆匀速行驶, 运行条件和环境良好, 车辆与平台保持互联互通.
- 3) 车辆完全遵从调度指令和路线行驶, 不考虑道路交通、天气、司机等外部因素的影响.
- 4) 忽略乘客上下车、车辆启停时间.
- 5) 忽略车辆加油或充电带来的影响.

1.2 点对点拼车聚类

在出行高峰、客流聚集等特殊时段, 乘客需求激增, 为了提高运营效率, 本文提出点对点拼车模式. 点对点拼车模式是一种基于乘客需求和车队资源优化匹配的出行方式, 该模式组合包车、拼车特点, 避免乘客绕路的同时降低出行成本.

平台根据该类型订单期望出发时间、乘客数、起始点和目的点, 基于 K -means 聚类算法, 对点对点拼车订单进行聚类, 构成点对点拼车订单组, 分配至同辆车, 车辆在中途不再匹配其他乘客, 从而实现点对点的共享出行, 具体方法如下:

- 1) 确定分类组数: 规划周期内, 计算所有点对点拼车订单的乘客人数 n_{sum} , 异质车队的平均载客量为 c' , 则分类组数定义为 $y = \left\lceil \frac{n_{\text{sum}}}{c'} \right\rceil + \beta$, 其中 β 为校准数, 取正整数.
- 2) 初始化: 订单的起始点和目的点构成数据点 $x_i = (o_i, d_i)$, 其构成数据矩阵 $\{x_1, x_2, \dots, x_i, \dots\}$, 从中随机选取 y 个样本作为初始聚类中心.
- 3) 更新聚类中心: 分别计算数据 x_i 与各聚类中心间的距离, 将其分配至距离最近的聚类中心, 根据分类结果更新每个类的聚类中心.
- 4) 更新分类结果: 计算新聚类中所有数据点的

均值, 得到新的聚类中心.

5) 终止条件: 重复执行步骤 3)、4), 直至每个数据点与其聚类中心的距离之和, 在前后两次迭代变化量小于设置阈值, 输出聚类结果.

6) 调整聚类结果: 判断聚类后每个聚类组的乘客总人数, 若组内乘客总数大于最大车型的可载客容量, 则对该聚类组进行进一步的聚类划分, 划分组数为 $y' = \left\lceil \frac{n'_{\text{sum}}}{c'} \right\rceil + \beta'$, 其中 n'_{sum} 为该聚类组总乘客数, β' 为新校准数, 取正整数.

根据上述步骤得出点对点拼车聚类组, 将同一聚类组内订单进行整合, 形成组合订单 $Q^s = \{n, o, d, t\}$, 其中, 起始点 o 和目的点 d 为聚类组的聚类中心, 乘客数量 n 为聚类组内订单总人数, 出发时间 t 为订单中最晚出发时间, 如图 2 所示, 生成新的起始点和目的点, 平台对聚类组进行匹配.

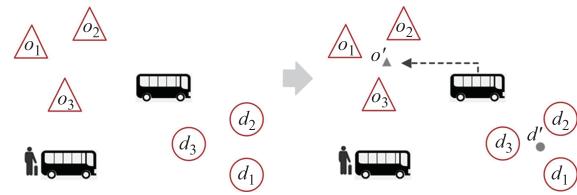


图 2 点对点拼车聚类

Fig.2 Point-to-point ride-sharing clustering

1.3 多车型混合运营模型

本文构建 MVTMO 模型, 乘客可自行选择出行方式, 即包车、拼车、点对点拼车三种模式. 平台根据订单信息及车辆信息, 以最大化运营效益为目标, 在时间窗、车辆容量等约束条件下, 对订单和车辆进行优化匹配, 满足个性化出行需求. 为了便于 MVTMO 模型建模, 本文将所有车辆、订单需求抽象为网络节点, 从而将其转换为网络流^[20]问题.

MVTMO 网络由节点和弧构成, 需求节点 $Q = \{Q^b, Q^p, Q^s\}$, 分别代表三种类型订单, 若两节点代表需求可由同一车辆完成, 则两节点相连. 对于任一时刻, 网络第一层为车辆节点 V , 最后一层为规划周期结束节点 S , 其余为需求节点 Q , 节点间根据时间、车辆容量等约束连接成弧, 如图 3 所示.

图 3 中, $V-Q$ 连接弧表示车辆需求匹配成功, $Q-Q$ 连接弧表示需求由同一车辆完成, 连接需满足式(1)~式(2), 其中 t_a 为到达时间, t_n 为乘客最大可忍受等待时长, C_v 为车辆载客量. $Q-S$ 连接弧表示完成需求后空载直至规划周期结束. 若 V 与 S 连接, 则

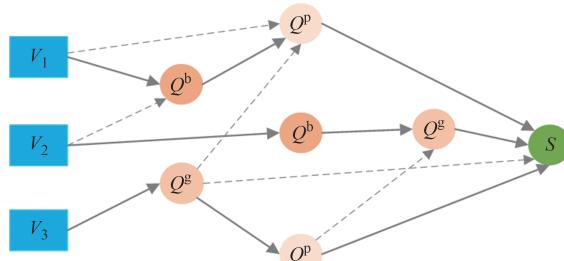


图 3 MVTMO 网络示意图

Fig.3 An illustration of MVTMO network

表示规划周期内车辆处于空载状态.

$$t_a \leq t + t_n \quad (1)$$

$$n \leq C_v \quad (2)$$

MVTMO 模型目标函数、约束条件定义如下:

1) 目标函数

由于车辆接客时间与订单期望出发时间可能存在出入,本文采用双层时间窗,定义惩罚函数.

对于包车、点对点拼车订单,如图 4 所示,第 i 个订单 Q_i 的期望出发时间为 t_{Qi} ,设置乘客最大可接受等待时长 t_m 和最大可忍受等待时长 t_n .当时间 $t < t_{Qi} + t_m$ 时,乘客直接等待车辆到达;如果 $t \in [t_{Qi} + t_m, t_{Qi} + t_n]$,乘客继续等待,但平台给予补偿;如果 $t > t_{Qi} + t_n$,则安排其他车辆服务.对于车辆而言,存在一个可等待乘客时长 t_c ,如果车辆到达起始点时间 $t \in [t_{Qi} - t_c, t_{Qi}]$,则车辆可接受等待;如果 $t < t_{Qi} - t_c$,则产生等待成本.

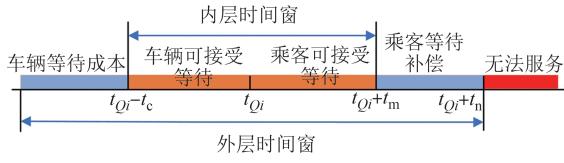


图 4 双层时间窗

Fig.4 Double-layer time window

包车、点对点拼车惩罚计算方式如式(3)所示,其中 f^v 为单位时间内车辆的等待成本, f^h 为单位时间的乘客补偿.

$$F = \begin{cases} f^v(t_{Qi} - t), & t < t_{Qi} - t_c \\ 0, & t_{Qi} - t_c \leq t \leq t_{Qi} + t_m \\ f^h(t - t_{Qi} - t_m), & t_{Qi} + t_m < t \leq t_{Qi} + t_n \end{cases} \quad (3)$$

对于拼车订单,本文考虑因车辆绕行的实际到达时间和期望时间差值,如图 5 所示,根据订单存在期望到达时间为 t_a ,设置最大可接受绕行时间 t_{11} 和最大绕行时间 t_1 ,内层时间窗时段 $[t_a, t_a + t_{11}]$ 不产生额外成本,其余时段产生补偿成本.

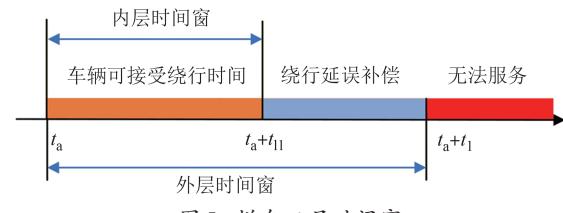


图 5 拼车双层时间窗

Fig.5 Ride-sharing double-layer time window

拼车惩罚计算方式见式(4):

$$F = \begin{cases} 0, & t_a \leq t \leq t_a + t_{11} \\ f^h(t - t_a - t_{11}), & t_a + t_{11} < t \leq t_a + t_1 \end{cases} \quad (4)$$

根据上述时间窗惩罚函数,本文以最大化运营效益 Z 为目标,提高需求服务率,同时保证服务质量,综合考虑运营商和乘客出行效益,具体定义如下:

$$\max Z = Z_b + Z_p + Z_g \quad (5)$$

$$Z_b = \sum_{v \in V} \sum_{i \in Q} \sum_{j \in Q} X_{ij}^v (f^b(o_j, d_j) - F_v \mathcal{D}(o_j, d_j) - F^b) \quad (6)$$

$$Z_p = \sum_{v \in V} \sum_{i \in Q} \sum_{j \in Q} X_{ij}^v (f^p(o_j, d_j) - F_v \mathcal{D}(o_j, d_j) - F^p d_i^v - F^p) \quad (7)$$

$$Z_g = \sum_{v \in V} \sum_{i \in Q} \sum_{j \in Q} X_{ij}^v (f^g(o_j, d_j) - F_v \mathcal{D}(o_j, d_j) - F^g) \quad (8)$$

式中: $f(o, d)$ 为订单支付费用; $\mathcal{D}(o, d)$ 为两节点之间的行驶距离; F_v 为车辆单位距离的运营成本; F^p 为单位距离绕行成本; d_i^v 为绕行距离.

2) 约束条件

考虑网络流问题和实际运营约束,MVTMO 模型约束条件包括网络流、节点、车辆载客、时间等.

网络流约束如式(9)~式(11)所示,式(9)表示车辆在规划周期内处于服务某一节点或空载状态.式(10)表示车辆在服务完节点 j 后可以选择服务其他节点或不服务任何节点.式(11)表示在规划周期结束时车辆节点或服务节点最终归于结束节点.

$$\sum_{i \in Q} X_{vi} + X_{vs} = 1, \forall v \in V \quad (9)$$

$$\sum_{i \in Q} \sum_{j \in Q} X_{ij}^v + X_{vj} = \sum_{j \in Q} \sum_{m \in Q} X_{jm}^v + X_{js}, \forall v \in V \quad (10)$$

$$\sum_{v \in V} X_{vs} + \sum_{v \in V} \sum_{j \in Q} X_{js}^v = V \quad (11)$$

节点约束如式(12)~式(14)所示,式(12)表示车辆只能从至多一个节点到达另一个节点.式(13)表示车辆从某一节点出发,至多只能到达一个节点.式(14)指每个服务节点最多被一辆车所服务.

$$\sum_{j \in Q} X_{jm}^v + X_{js} \leq 1, \forall v \in V, m \in Q \quad (12)$$

$$\sum_{j \in Q} X_{ij}^v + X_{sj} \leq 1, \forall v \in V, i \in Q \quad (13)$$

$$\sum_{v \in V} \sum_{i \in Q} (X_{ij}^v + X_{vi}) \leq 1 \quad (14)$$

车辆载客约束如式(15)~式(17)所示,式(15)表示包车、点对点拼车订单载客约束, C_v 为车辆最大容量.式(16)、式(17)表示拼车订单载客约束,式(16)为车辆载客量变化,式(17)表示车辆载客量应不超过最大容量, N^v 为节点车辆载客数.

$$X_{ij}^v \cdot n \leq C_v, \forall v \in V, i, j \in Q \quad (15)$$

$$X_{ij}^v \cdot (N_i^v + n_i) = X_{ij}^v \cdot N_j^v, \forall v \in V, i, j \in Q \quad (16)$$

$$N_i^v, N_j^v \leq C_v, \forall v \in V, i, j \in Q \quad (17)$$

时间约束如式(18)~式(20)所示,式(18)表示车辆到达服务节点起点时间不大于服务节点的最大可忍受等待时间, $t(d_i, o_j)$ 表示两节点之间的行驶时间.式(19)表示拼车订单最大绕行时间限制.

$$X_{ij}^v [t + t(d_i, o_j)] \leq t_j + t_n, \forall v \in V, i, j \in Q \quad (18)$$

$$t_j^a \leq t_j + t_n, \forall j \in Q \quad (19)$$

模型决策变量范围如式(20)所示.

$$X_{ij}^v, X_{vi}, X_{vs} \in \{0, 1\} \quad (20)$$

2 求解方法

在本文所提 MVTMO 问题中,模型求解复杂度较高,且随着问题规模扩展而显著增大.因此,本文采用 ALNS 算法,根据问题特点优化邻域搜索操作,满足大规模问题的高效求解.

2.1 算法流程

ALNS 算法的主要流程为:以贪婪策略构造初始解;设置初始参数,包含算子初始得分、接受准则初始温度等,初始解设置为当前解与最优解;采用轮盘赌法选择移除、插入算子,对当前解进行相应操作,生成新解,同时更新算子使用次数;判断新产生解是否被接受,并更新算子得分;每隔一定迭代次数更新算子权重,并初始化算子得分、使用次数;基于终止条件结束算法,输出最优解. ALNS 算法主要流程如图 6 所示.

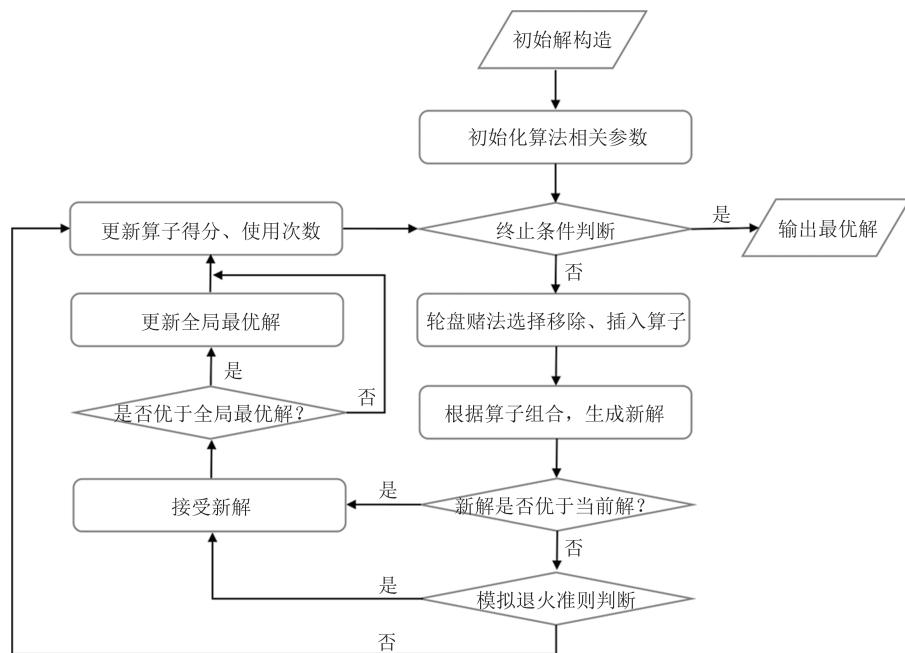


图 6 ALNS 算法主要流程

Fig.6 Main process of ALNS algorithm

2.2 初始解构造

初始解构造基于距离最近策略,即为待匹配订单分配车辆.待匹配订单包括新产生订单和灵活订单.平台根据期望出发时间对待匹配订单进行排序,按照顺序依次安排车辆以构成初始解.对于每个待匹配订单,基于距离最近原则,若满足时间窗、容量

等约束,则匹配成功,并将行程添加至车辆路径;若不满足约束,则寻找距离次近车辆进行相同判断.若所有车辆均不满足约束,则订单响应失败.

2.3 邻域搜索

邻域搜索直接影响算法的搜索效率和解的质量,因此,针对 MVTMO 特性,设计了三种不同的移

除、插入算子，并设计了相应权重更新策略。

首先设计移除算子，移除算子是从当前解中移除 a 个订单，设匹配成功订单总数为 l ，引入最大移除系数 γ ，即移除订单数不超过总数的 γ 倍。本文采用三种移除算子，根据移除算子定义，对当前解进行移除操作，从而生成新解，算子具体定义如下：

1) 随机移除：随机移除算子指在当前解中随机地选取订单进行移除操作，每个订单均有概率被移除，移除订单之间不存在关联。

2) 最差移除：最差移除算子指在当前解中选取收益影响最小订单进行移除，引入参数 p_q^r ，即订单 q 移除路径 r 所导致的收益减少值，计算公式如下：

$$p_q^r = F(r) - F_{-q}(r) \quad (21)$$

式中： $F(r)$ 为路径 r 原本收益； $F_{-q}(r)$ 为路径 r 移除订单 q 后收益。按 p_q^r 大小升序排序，选取前 a 个订单进行移除。

3) 相似移除：相似移除算子通过移除相似订单，以期获得更优解。订单相似性通常考虑行程距离、时间窗等因素，由于不同类型订单间的乘客数量差异较大，因此本文在订单相似性判断中额外考虑乘客数量因素，计算如下：

$$L(i, j) = \psi [d(o_i, o_j) + d(d_i, d_j)] + \mu |t_i - t_j| + \omega (n_i - n_j) \quad (22)$$

式中： $L(i, j)$ 为订单 i 和 j 的相似度； ψ 、 μ 和 ω 分别为订单行程、时间窗和乘客数量对应权重。 $L(i, j)$ 越小，表示相似性越高。在待移除订单中随机选取一个订单 i ，计算其余匹配成功订单与 i 的相似度，并按 $L(i, j)$ 大小进行升序排列，选取首个订单移除。

其次设计插入算子，插入算子是将被移除订单根据一定的规则重新插入车辆路径。本文采用随机插入、贪婪插入和 Regret-2 插入算子对被移除订单进行重分配与插入，具体定义如下：

1) 随机插入：随机插入算子指在被移除订单集合中随机选取，并随机选择车辆与之匹配，根据约束条件下将其插入车辆路径。若不满足约束，则重新选择车辆，直至所有被移除订单均插入。

2) 最优插入：最优插入算子通过选择最大收益增量的订单插入路径，收益增量 u_q^r 计算公式如下：

$$u_q^r = F_{+q}(r) - F(r) \quad (23)$$

式中： $F(r)$ 为路径 r 原本收益； $F_{+q}(r)$ 为路径 r 插入订单 q 后总收益。计算所有车辆与订单匹配产生的收

益增量，最大收益增量 $u_q = \max \{u_q^r\}$ ，根据 u_q 确定所分配订单 q ，与相应车辆进行匹配，直到所有被移除订单均插入。

3) Regret-2 插入：Regret-2 插入算子计算每个待插入订单插入不同车辆路径的收益增量，降序排列。最大收益增量 $u_q^{r_1}$ 为最优插入， $u_q^{r_2}$ 为次优插入收益增量，则后悔程度 u_q^* 计算公式如下：

$$u_q^* = u_q^{r_1} - u_q^{r_2} \quad (24)$$

u_q^* 越大，插入优先级越高。在每次迭代中，计算所有订单的 u_q^* ，将 u_q^* 最大所对应的订单 q 与相应车辆进行匹配，直到所有被移除订单均被插入。

ALNS 的自适应体现操作算子的权重更新，为了选取合理移除插入组合，本文通过轮盘赌法确定选择每个算子的概率，被选中概率为 $p = \frac{w_i}{\sum w_i}$ ， w_i 为算子权重。

初始化时，各算子权重相同，之后根据算子表现和使用次数对算子权重进行更新。算子表现体现为对算子进行打分，将算法迭代过程均匀划分一定轮次，在各轮次中，根据算子所产生的解的质量进行打分，三种分为 $\sigma_1, \sigma_2, \sigma_3$ ，定义如下：

1) 算子在本轮次中产生新的全局最优解，邻域操作性能良好，给予较高分数 σ_1 ；

2) 算子在本轮次中产生优于当前解的新解，即算子推动了解的优化，给予次高分数 σ_2 ；

3) 算子在本轮次中产生的解次于当前解，但依据 Metropolis 准则仍被接受，有利于加强对解空间的探索，给予分数 σ_3 ；

4) 本轮次中未出现上述三种情况，表明当前操作对寻优过程贡献有限，得分不发生改变。

设置每隔 n 次迭代，根据算子在迭代轮次内的分数累加和使用次数，对算子权重进行更新，计算公式如下：

$$w_i = \begin{cases} w_i(1 - \varepsilon) + \varepsilon \frac{S_i}{\theta_i}, & \theta_i > 0 \\ w_i, & \theta_i = 0 \end{cases} \quad (25)$$

式中： S_i 为算子在该轮次中总分数； θ_i 为算子在该轮次中被使用次数； ε 为调节权重系数，用于调节该轮次算子分数和使用次数对权重影响的大小。

2.4 接受准则与终止条件

接受准则和终止条件影响着解的质量和求解时间，为此引入模拟退火的 Metropolis 准则^[21]，避免陷

入局部最优,保持算法在较差解的基础上得到最优解的可能性,即假设当前解的目标函数值为 Z_{cur} , 新解的目标函数值为 Z_{new} , 则较差新解 Z_{new} 被接受的概率如下:

$$p = \exp\left(\frac{Z_{\text{new}} - Z_{\text{cur}}}{T_k}\right) \quad (26)$$

式中: T_k 为当前迭代轮次 k 的模拟退火温度, 即新解被接受的概率取决于目标函数值的差与当前时刻模拟退火的温度 T_k 的比值. T_k 的计算公式如下:

$$T_k = T_0 \cdot c^k \quad (27)$$

式中: T_0 为初始温度; c^k 为冷却率系数.

由此可见, 随着迭代推进, 模拟退火的温度逐渐减小, 即接受较差解概率降低, 算法趋于收敛. T_0 的设置如下:

$$T_0 = \frac{-\tau}{\ln 0.5} |Z_0| \quad (28)$$

式中: $0 < \tau < 1$, 即相较于初始解, 存在劣质比率 τ 的较差解被接受概率为 50%.

最后, 为在适当时刻终止迭代, 本文以模拟退火准则以及最优解的连续未改进次数 m 作为终止条件. 首先, 算法终止需满足相较于当前解, 存在 τ 的较差解被接受的概率小于 ϑ , 即:

$$T_{\min} < \frac{-\tau}{\ln \vartheta} |Z_{\text{cur}}| \quad (29)$$

在此基础上, 若算法仍存在连续 m 次迭代未改进, 则终止迭代过程.

3 试验验证与分析

本节针对 MVTMO 模型以及 ALNS 算法进行分析, 首先验证改进算法性能, 其次与参考模型对比, 同时验证模型针对不同场景的适应性, 最后对模型参数进行敏感性分析. 试验在 Python 3.10 环境下运行, 计算机配置为 Intel (R) Core (TM) i7-9750H CPU@2.60 GHz, 8.0 GB 内存.

3.1 试验设置

为充分验证所提出模型及其求解算法, 本文基于武汉市某区域真实道路场景, 根据实地调研的需求数据, 选择 3 个停车点, 分别靠近地铁站、居民区、企业区, 车辆完成任务后前往就近停车点.

在车队配置方面, 考虑到不同类型订单的运力需求, 本文选取三种车型, 根据现有研究设置车型的相关配置参数. 运力方面相关信息如表 1 所示, 车型系数影响运营成本.

表 1 车型配置信息

Tab.1 Vehicle configuration of driving types

车型	等待费用/(元·min ⁻¹)	最大载客量/人	车型系数
1	0.2	4	1.0
2	0.28	7	1.6
3	0.35	10	2.1

在订单需求方面, 对实际订单数据进行处理. 考虑到同一订单的乘客数可能存在较大差异, 所以根据乘客数对订单设置修正系数 α , $\alpha \in \{1, 1.6, 2.4\}$, α 的大小与乘客数量正相关. 通过 α 对需求参数进行修正, 如表 2 所示.

表 2 乘客需求参数设置

Tab.2 Parameters setting of passengers request

订单类型	等待补偿/(元·min ⁻¹)	可接受等待或延误时长/min	可忍受等待或绕行时长/min
包车	0.4 α	4	7
拼车	0.34 α	3	5
点对点拼车	0.3 α	5	7

ALNS 算法相关参数如表 3 所示.

表 3 ALNS 参数设置

Tab.3 Parameters setting of ALNS

参数符号	参数定义	取值
γ	最大移除系数	0.4
ψ	行程权重	3.0
μ	时间窗权重	1.0
ω	乘客数量权重	2.0
σ_1	全局最优评分	3.0
σ_2	当前最优评分	2.0
σ_3	接受较差解评分	1.0
ε	调节权重系数	0.3
c	冷却率	0.99
τ	劣质比率	0.05
ϑ	较差解接受概率	0.01

3.2 算法性能分析

为了验证本文改进的 ALNS 算法性能, 采用两种对比算法:

1) NVD(nearest vehicle dispatch) 算法, 即对于待分配的订单, 找到距离其最近车辆, 根据车辆已规划行驶路径, 在满足容量、时间等约束下, 寻找合适车辆并插入行驶路径;

2) BIS(best insertion search) 算法, 即将所有待分配的订单按期望出发时间先后进行处理, 依据容量、时间等约束, 找出订单可匹配所有车辆, 并依次

计算各车辆收益增加值,将订单匹配至收益增加值最大车辆.

如图 7 所示,改进后的 ALNS 算法与 NVD 算法、BIS 算法在不同车队规模下对比指标性能.由图 7 可以看出,当车队规模较小时,ALNS 算法的求解优势不明显.随着车队规模的增大,求解问题的规模变大,ALNS 算法的求解优势逐渐明显,各方面指标性能提升.当车队规模达到 90 辆时,求解质量对比 NVD、BIS,在运营总收益方面分别提升 19%、12%,订单响应率方面提升 15%、7%,乘客平均等待时间下降 42%、33%,空载行驶率下降 24%、16%,效果显著.这是因为 ALNS 算法基于贪婪算法构造的初始解基础,通过不断地执行订单移除和插入操作,并加入模拟退火准则进行解的取舍,更有利于寻找全局最优解,克服 NVD 和 BIS 的短视性问题.所以,改进后的 ALNS 算法能有效求解大规模问题.

3.3 模型分析

为验证 MVTMO 模型,采用以下两种模型进行对比:

1) 优化再分配模型^[22] (optimization-reassignment model, ORM):基于距离最近策略,在考虑待分配请求和空闲车辆基础上,进一步考虑对已匹配但尚未上车请求的重分配;

2) 按需拼车^[23] (on-demand ride-sharing, ODRS):根据在线需求和车辆位置,基于贪婪分配策略,通过不断优化对路线进行改进,生成高质量行驶路径.

如图 8 所示,MVTMO 与 ORM、ODRS 在不同车队规模下,对不同指标进行对比.当车队规模较小时,可用车辆少,且拼车订单受时间窗约束的限制,MVTMO 订单响应率受到影响,导致性能提升效果较差.随着车队规模增大,MVTMO 通过对包车、拼车、点对点拼车订单灵活匹配,性能提升效果逐渐明显.当车队规模达到 100 辆时,对比 ORM、ODRS,在运营总收益分别提升 9%、7%,订单响应率提升 10%、6%,平均等待时间下降 30%、25%,空载行驶率下降 15%、8%.由此可见,MVTMO 随着车队规模增大,提升效果增加,适用于较大规模车队的灵活调度.

此外,将早上 7 点至 9 点和晚上 6 点至 8 点定为高峰时段,其余为平常时段,将车辆运营模式划分为包车、拼车模式,即车辆在该时段只服务包车(拼车)

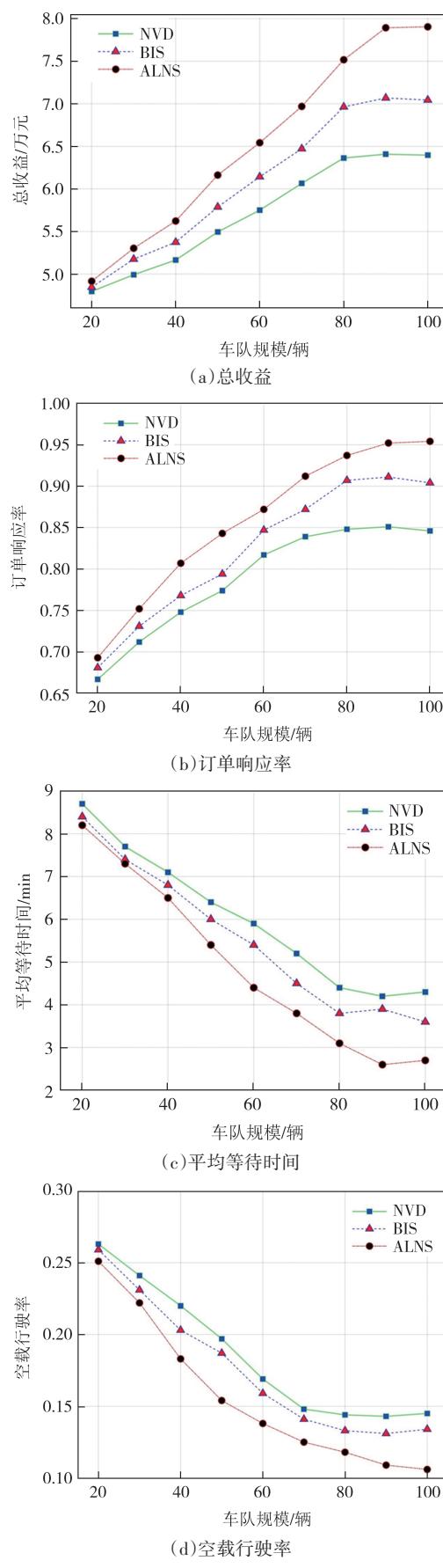


图 7 算法性能分析

Fig.7 Performance analysis of algorithm

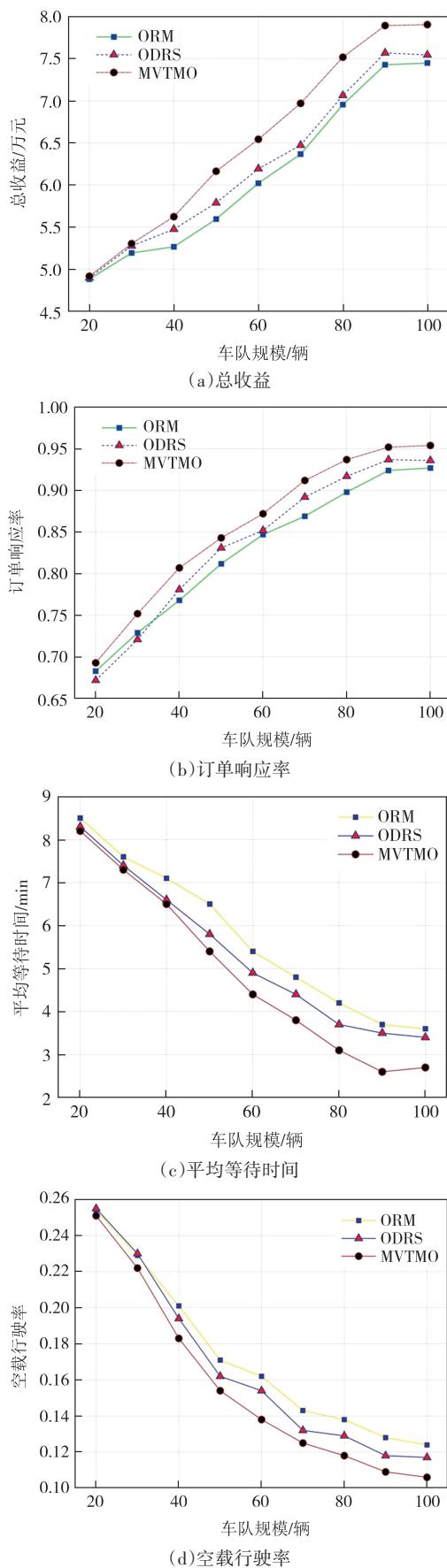


图 8 模型性能分析

Fig.8 Performance analysis of model

类型订单.设置不同模式的车辆比例,分析MVTMO面对不同时段的适应性,对比MVTMO与ORM、ODRS在不同时段的性能指标,如表4~表5所示.

表4 平常时段适应性分析

Tab.4 Adaptability analysis in normal periods

模型	比例	运营收益/万元	订单响应率/%	平均等待时间/min	空载行驶率/%
MVTMO	1:1	43 652	94.1	2.7	11.8
	1:2	41 024	93.7	3.1	11.7
	2:1	45 975	95.2	2.6	12.0
ORM	—	43 875	94.5	2.9	12.4
ODRS	—	39 674	92.3	3.3	11.4

表5 高峰时段适应性分析

Tab.5 Adaptability analysis in peak periods

模型	比例	运营收益/万元	订单响应率/%	平均等待时间/min	空载行驶率/%
MVTMO	1:1	36 079	95.9	2.4	9.5
	1:2	38 385	97.1	2.5	9.1
	2:1	35 256	95.2	2.7	9.3
ORM	—	35 637	95.4	3.0	9.6
ODRS	—	37 475	96.3	2.7	9.2

由表4可以看出,在运营周期的平常时段内,MVTMO不同比例下所得结果相近,当比例为2:1时,即处于包车模式的车辆更多,对比其余情况,虽然空载行驶率较大,但运营收益、订单响应率、平均等待时间均有所改善,这是因为在运营周期的平常时段内,乘客订单需求较为分散,拼车类型订单受到时间窗、最大绕行时间等约束限制,匹配成功率低,影响需求的响应率.因此可在该时段设置更多车辆服务于包车模式,以此有效响应分布式乘客需求.在此比例下,将MVTMO与ORM、ODRS进行对比,ODRS受到时间窗、绕行时间等因素限制,订单响应率较低,影响运营收益.ORM能服务较多订单,但该模型缺少拼车类型服务,对比MVTMO,ORM平均等待时间与空载行驶率较大,影响运营收益与乘客出行质量.可以看出,MVTMO在运营的平常时段性能更优.

由表5可以看出,在运营高峰时段内,设置更多车辆处于拼车服务模式,MVTMO的性能更优,运营收益、订单响应率高,且能减少等待时间,提升乘客的出行质量.这是因为高峰时段乘客的订单需求较为集中,设置更多车辆服务于拼车服务模式,能在一定时间内响应更多的订单.因此在运营周期的高峰时段内,可以设置更多的车辆处于拼车服务模式.

在此比例下,将 MVTMO 与 ORM、ODRS 进行对比,ORM 只采用包车服务,其车辆利用率较低,影响订单响应率,导致运营收益较低.ODRS 由于高峰时段的需求集中特性,订单响应率提升,但受到车辆容量以及绕行时间限制,ODRS 的订单响应率低于 MVTMO, 乘客平均等待时间较大, 所以 MVTMO 在运营的高峰时段性能更优.

总之,MVTMO 优于 ORM、ODRS, 能有效适应不同时段. 平台可以根据运营时段灵活调整服务模式, 实现异质车队的灵活配置.

3.4 敏感性分析

首先,分析包车、拼车、点对点拼车三种类型订单数量比对 MVTMO 模型的影响.

如表 6 所示,随着拼车和点对点拼车类型订单增多,订单响应率提高,这是因为订单间具有相似行程的概率更高,从而更容易匹配成功,提高车辆平均载客数,车辆座位利用率提升. 因此,当车辆存在供应不足情况,可以通过鼓励乘客选择拼车或点对点拼车的出行方式,提高订单响应率和车辆利用率,提升乘客和平台双方效益.

表 6 订单类型比值影响

Tab.6 Effect of order type ratio

订单类型比	订单响应率/%	车辆平均载客数	空载行驶率/%
2:1:1	95.2	2.7	10.8
1:1:1	94.8	3.0	10.3
1:2:1	96.1	3.8	9.9
1:1:2	95.7	3.7	10.1

其次,研究异质车队车型配比的影响. 如表 7 所示,当车型配比设置为 1:1:1 时,产生的运营收益最大. 随着大容量车型占比增加,订单响应率、空载行驶率变化较小,但是运营收益有所下降,这是因为大容量车型运营成本和惩罚成本更高,乘客数较少时,与大容量车型匹配会造成运营成本增加,影响车辆利用率. 因此在实际运营过程中,可根据不同类型订单数量波动,对车型配比进行调整. 当包车订单较多时,增加小型车使用比例,当拼车和点对点拼车订单较多时,增加中型车使用比例.

最后,对拼车订单的最长绕行时间 t_1 进行分析,研究 t_1 对模型的影响.

如表 8 所示,受到 t_1 的影响,当设置较小的 t_1 时,平均绕行距离较小,乘客出行质量高,但是较小的 t_1 影响 MVTMO 的拼车订单响应,导致订单响应

表 7 异质车队车型配比影响

Tab.7 Effect of vehicle type ratio in a heterogeneous fleet

车型配比	运营收益/元	订单响应率/%	空载行驶率/%
2:1:1	79 120	95.5	10.6
1:1:1	79 567	94.8	10.2
1:2:1	78 964	95.7	9.8
1:1:2	78 673	95.8	10.1

率低. 当设置较大的 t_1 时,更多的拼车订单得到服务,订单响应率和运营收益高,但是绕行时间增大的同时导致绕行距离增加,影响乘客的出行体验. 所以 MVTMO 可根据运营收益和乘客出行质量权衡,按实际需求设置 t_1 的大小.

表 8 最长绕行时间设置

Tab.8 Setting of maximum detour time

最长绕行 时间 t_1	运营收益/元	订单响应率/%	平均等待 时间/min	平均绕行 距离/km
3	77 456	92.8	2.8	0.743
4	78 064	94.2	3.0	0.967
5	79 476	95.8	3.2	1.127
6	79 521	95.7	3.1	1.164
7	78 978	96.1	3.5	1.243

4 结 论

针对需求响应公交服务,本文统筹考虑包车、拼车、点对点拼车三种运营模式,构建包含异质车队的多车型混合运营模型,实现车队的灵活调度与配置,以此提升运营商与乘客双方效益. 此外,为了保证求解速度和求解质量,设计基于 ALNS 的求解算法. 之后,对多车型混合运营模型及求解算法进行分析,结果表明:随着车队规模的逐渐增大,改进后的 ALNS 算法求解效果显著提升. 其次,多车型混合运营模型对比包车、拼车模型,随着车队规模增大,性能显著提升. 当车队规模达到 100 辆时,运营收益方面分别提升 9%、7%,平均等待时间下降 30% 和 25%,综合提升运营效益和乘客出行体验. 多车型混合运营模型可通过调整服务模式,灵活调整车队配置,以应对特殊时段需求波动. 敏感性分析表明所提模型能根据参数变化灵活调整,确保模型有效性和适应性.

然而,本文未考虑乘客取消订单或者司机拒单的情况,同时缺少对司机收入考虑. 在未来研究中,可以综合考虑运营商、乘客、司机三方的收益,改进模型目标函数以及相关参数定义. 同时,进一步考虑

模型处理不同情况城市路网、不同乘客需求的稳定性,以此验证模型的实用性.

参考文献

- [1] XU Y H, WANG W Y, XIONG G W, et al. Network-flow-based efficient vehicle dispatch for city-scale ride-hailing systems [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 23(6):5526–5538.
- [2] DING H, LI J Y, ZHENG N, et al. Dynamic dispatch of connected taxis for large-scale urban road networks with stochastic demands: an MFD-enabled hierarchical and cooperative approach [J]. *Transportation Research Part C: Emerging Technologies*, 2022, 142:103792.
- [3] GAO J, LI X M, WANG C, et al. BM-DDPG: an integrated dispatching framework for ride-hailing systems [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 23(8): 11666–11676.
- [4] VAN ENGELEN M, CATS O, POST H, et al. Enhancing flexible transport services with demand-anticipatory insertion heuristics [J]. *Transportation Research Part E: Logistics and Transportation Review*, 2018, 110:110–121.
- [5] LI S, ZHONG Z H, SHI B. Ride-hailing order matching and vehicle repositioning based on vehicle value function [C]// Knowledge Science, Engineering and Management. Cham: Springer, 2022: 406–416.
- [6] MESKAR M, ASLANI S, MODARRES M. Spatio-temporal pricing algorithm for ride-hailing platforms where drivers can decline ride requests [J]. *Transportation Research Part C: Emerging Technologies*, 2023, 153:104200.
- [7] CHEN X Q, CHEN X W, ZHENG H Y, et al. Efficient dispatching for on-demand ride services: systematic optimization via Monte-Carlo tree search [J]. *Transportation Research Part C: Emerging Technologies*, 2021, 127:103156.
- [8] YOU J P, JIANG H W, CHEN Z Y, et al. Order allocation strategy for online car-hailing platform in the context of multi-party interests [J]. *Advanced Engineering Informatics*, 2023, 57: 102110.
- [9] GUO Y H, LI W H, XIAO L F, et al. Enhancing efficiency and interpretability: a multi-objective dispatching strategy for autonomous service vehicles in ride-hailing [J]. *Computers & Industrial Engineering*, 2024, 194:110385.
- [10] SHI B, XIA Y M, XU S, et al. A vehicle value based ride-hailing order matching and dispatching algorithm [J]. *Engineering Applications of Artificial Intelligence*, 2024, 132:107954.
- [11] HUANG J X, LUO Y C, XU M, et al. pShare: privacy-preserving ride-sharing system with minimum-detouring route [J]. *Applied Sciences*, 2022, 12(2): 842.
- [12] MESHKANI S M, FAROOQ B. A generalized ride-matching approach for sustainable shared mobility [J]. *Sustainable Cities and Society*, 2022, 76: 103383.
- [13] TUNCEL K, KOUTSOPoulos H N, MA Z L. An integrated ride-matching and vehicle-rebalancing model for shared mobility on-demand services [J]. *Computers & Operations Research*, 2023, 159:106317.
- [14] LI X F, DU M Y, KWAN M P. Order dispatch optimization with considering flexible one-to-three matching strategy under hybrid ride-hailing service modes [J]. *Transportation Letters*, 2023, 15(10): 1337–1353.
- [15] SIMONETTO A, MONTEIL J, GAMBELLA C. Real-time city-scale ridesharing via linear assignment problems [J]. *Transportation Research Part C: Emerging Technologies*, 2019, 101: 208–232.
- [16] GUO Y H, ZHANG Y, BOULAKSIL Y. Real-time ride-sharing framework with dynamic timeframe and anticipation-based migration [J]. *European Journal of Operational Research*, 2021, 288(3): 810–828.
- [17] GHANDEHARIOUN Z, KOUVELAS A. Real-time ridesharing operations for on-demand capacitated systems considering dynamic travel time information [J]. *Transportation Research Part C: Emerging Technologies*, 2023, 151: 104115.
- [18] HYLAND M, MAHMASSANI H S. Operational benefits and challenges of shared-ride automated mobility-on-demand services [J]. *Transportation Research Part A: Policy and Practice*, 2020, 134:251–270.
- [19] YAO R, BEKHOR S. A dynamic tree algorithm for peer-to-peer ridesharing matching [J]. *Networks and Spatial Economics*, 2021, 21(4):801–837.
- [20] VAZIFEH M M, SANTI P, RESTA G, et al. Addressing the minimum fleet problem in on-demand urban mobility [J]. *Nature*, 2018, 557(7706):534–538.
- [21] ZHONG Y W, LIN J, WANG L J, et al. Discrete comprehensive learning particle swarm optimization algorithm with Metropolis acceptance criterion for traveling salesman problem [J]. *Swarm and Evolutionary Computation*, 2018, 42: 77–88.
- [22] HYLAND M, MAHMASSANI H S. Dynamic autonomous vehicle fleet operations: optimization-based strategies to assign AVs to immediate traveler demand requests [J]. *Transportation Research Part C: Emerging Technologies*, 2018, 92:278–297.
- [23] ALONSO-MORA J, SAMARANAYAKE S, WALLAR A, et al. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment [J]. *PNAS*, 2017, 114(3):462–467.