

考虑骑手异质性的众包配送策略优化^①

王会静, 袁鹏程

(上海理工大学 管理学院, 上海 200093)

通信作者: 袁鹏程, E-mail: danis_cx@126.com



摘要: 针对当前众包平台面临的订单类型多样性 (外卖订单与快递订单) 和配送骑手的同质化 (单一外卖型与单一快递型) 问题, 且现有众包配送机制较少兼顾商家和顾客满意度, 在派单模式下考虑骑手的异质性, 通过引入全能型骑手, 将骑手划分为单一外卖型、单一快递型和全能型 3 类, 根据各类骑手可服务的订单类型差异, 构建基于模糊时间窗的商家和顾客对于骑手到达取送货点时间的满意度成本函数, 将商家和顾客的满意度转化为时间惩罚函数, 建立了以时间惩罚成本、路径行驶成本、人员运营成本最小化为目标的模型, 针对模型的特点以及传统算法存在的问题, 设计将遗传算法与大规模领域搜索算法相结合的混合算法, 随后通过具体算例, 采用模拟退火算法、遗传算法和混合算法分别进行求解, 通过不同算法的优化结果对比分析, 验证模型和改进算法的可行性和有效性. 实验结果表明: 在众包配送过程中合理考虑骑手异质性以及商家和顾客的满意度不仅能够有效提升商家和顾客满意度, 也能够降低众包平台配送成本和提高配送效率, 对于众包平台制定配送策略具有一定的参考意义.

关键词: 骑手异质性; 时间满意度; 订单分配; 路径优化; 混合算法

引用格式: 王会静, 袁鹏程. 考虑骑手异质性的众包配送策略优化. 计算机系统应用. <http://www.c-s-a.org.cn/1003-3254/9726.html>

Optimization of Crowdsourcing Delivery Strategy Considering Rider Heterogeneity

WANG Hui-Jing, YUAN Peng-Cheng

(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: In crowdsourcing platforms, orders have different types (takeaway and express orders), while delivery riders are typically responsible for only one type of order (either takeaway or express delivery). Additionally, the existing delivery mechanism rarely meets the satisfaction of merchants and customers. Therefore, considering the heterogeneity of riders in a dispatch mode, this study introduces the concept of all-round riders, dividing riders into three categories: takeaway riders, express riders, and all-round riders. According to the differences in the types of orders that riders can serve, a cost function based on a fuzzy time window is constructed to represent the satisfaction of merchants and customers with the time when riders arrive at pick-up and delivery points. The satisfaction is then transformed into a time penalty function. A model is constructed to minimize time penalty costs, route driving costs and personnel operation costs. Considering the characteristics of the model and the limitations of traditional algorithms, this study designs a hybrid algorithm combining genetic algorithms and search algorithms in large domains. Then, the simulated annealing algorithm, genetic algorithms, and hybrid algorithm are used to solve the problem respectively through concrete examples. The analysis of the optimization results of different algorithms validates the feasibility and effectiveness of the proposed model and the improved algorithm. Experimental results show that considering the heterogeneity of riders and the satisfaction of merchants and customers during crowdsourcing delivery not only effectively improves their satisfaction but also reduces delivery costs and improves delivery efficiency for crowdsourcing platforms. This strategy offers a reference

^① 基金项目: 国家自然科学基金 (71601118)

收稿时间: 2024-05-11; 修改时间: 2024-06-28; 采用时间: 2024-07-18; csa 在线出版时间: 2024-10-31

for crowdsourcing platforms in formulating delivery strategies.

Key words: rider heterogeneity; time satisfaction; order allocation; path optimization; hybrid algorithm

在当前社会,随着共享经济、电子商务和互联网技术的蓬勃发展,线上外卖和快递业务已经渗透到人们日常生活的方方面面.社会生活节奏的加快,使得用户对服务质量的需求持续攀升.与此同时,由于线上业务规模的扩大,小批量、小质量、多频次的配送需求呈爆发式增长,传统专业配送在配送价格、服务质量、时效性等方面面临着前所未有的考验^[1].为了降低物流企业的配送成本,同时满足特定时段激增的订单量,一种基于共享经济理念的众包配送模式被提出并得到了应用^[2].

基于众包的配送模式通常指的是众包配送平台将配送任务合理地分配给社会闲散的人员,在降低物流企业成本,减少资源浪费的同时也缓解高峰期的配送压力.常见的众包配送模式可分为平台派单模式和骑手抢单模式.在抢单模式下,众包平台在接收到订单后将这一时刻产生的所有订单放入抢单池中,众包骑手根据自身的能力、喜好自主进行抢单;在派单模式下,众包平台直接分配订单给众包骑手并规划路线.目前众包平台更倾向于发展派单模式并逐渐将抢单模式转变为派单模式.

在当前众包配送领域的研究中,大多研究集中于单一类型的骑手参与配送活动,具体而言,即外卖骑手仅负责外卖订单的配送,快递骑手仅处理快递订单.然而,当多个不同类型的订单在地理位置相近的地方出现时,通常会导致对更多骑手的需求,从而造成人力资源的过度投入与浪费;与此同时,众包配送过程中参与的主体有平台、商家、顾客和骑手,各方利益密切相关且相互影响,现有研究聚焦于平台、顾客和骑手对于服务质量和配送效率的提升,缺少对于商家满意度的考虑.因此设计考虑骑手异质性,满足高配送效率、高商家和顾客满意度的众包订单分配与路径规划方案,是本文要解决的问题.

1 相关研究

众包配送问题可以看作是订单分配与路径规划的组合优化问题^[3].

对于订单分配问题,许多学者进行了研究,Arslan

等^[4]研究如何使用剩余的社会资源,来构建包裹配送和临时众包配送匹配的服务平台,在此基础上设计一种基于滚动视图的建模方法以解决实时分配的动态拾取和交付问题.王爽等^[5]提出一种众包车辆和物流企业合作的新模式,在此基础上制定“众包司机-订单”匹配策略,设计众包司机配送方案.邓娜等^[6]从订单分配中的任务分配环节出发,提出了一种结合聚类分析和最小距离匹配原则的订单集指派模式.戴韬等^[7]从订单选择及订单执行路径出发,设计双层算法来比较接受不同订单所带来的收益,从而进行订单分配.冯爱兰等^[8]考虑在抢派结合的模式下,设计离散选择模型模拟骑手抢单决策,提高骑手对于订单的自主选择性.刘春玲等^[9]基于众包配送时效性问题,通过动态筛选订单配送方,结合冷链配送任务设计模糊机会策略进行订单分配.

对于路径规划问题,车辆路径问题作为物流配送中的关键环节,是当前相关研究中的热点^[10].众包配送问题具有多骑手、时间窗约束、先取后送和开环等特征,可以看作是带时间窗的取送货车辆路径问题.许多学者对此类问题进行了研究,陈星光等^[11]从配送员的角度出发,以配送员总收益最大化、配送总时长最小化为目标建立车辆路径模型,考虑外卖配送员依据偏好选取订单的决策行为,在此基础上设置配送系统优化策略. Ren 等^[12]通过将商家和顾客的满意度转化为惩罚函数来构建外卖配送的取送货车辆路径模型,旨在将总配送成本降至最低.孙云霄等^[13]考虑到配送员由于等级不同带来的载重能力和运营成本的不同,将众包配送员等级与最大承单量、额外成本相结合,求解取送货交叉的路径优化问题.刘烁佳等^[14]通过引入时间窗参数,研究如何调整客户的优先级来减少车辆的等待时间,建立了带时间窗的车辆路径优化模型.赵佳欣等^[15]从客户角度出发,考虑了不同载重、不同速度的多车型,建立了时变网络下同时取送货车辆的车辆路径优化模型.

在特定的情况下,通过对多个目标进行集成优化,可以对问题进行更全面的思考,并取得比较好的结果.因此,学者们将取送货问题的研究推广至多目标优化

领域. 张力娅等^[16]考虑到众包配送的实际流程, 提出了顾客的优先等级, 基于顾客满意度最大化和平台成本最小化为优化目标建立了众包配送车辆路径问题模型. 熊浩等^[17]从平台、骑手和顾客 3 个角度出发, 建立多目标外卖配送路径优化模型, 解决动态场景下多目标的外卖配送问题. 陈高华等^[18]针对配送过程中产生的配送成本和碳排放量, 考虑车辆的不同速度和载重负荷, 提出了同时考虑碳排放量和配送总成本的多目标车辆路径优化模型, 旨在解决带时间窗和容量约束的车辆路径问题.

在求解算法方面: Ibrahim 等^[19]在交叉和变异操作中考虑基因序列之间的距离, 设计了一种改进的遗传算法, 旨在找到一种解决 VRPTW 问题的最佳方案. 田亮等^[20]对送达时间不在时间窗口内的车辆采取一定的时间惩罚措施, 将带软时间窗的节约算法与时间惩罚成本进行结合, 设计改进的节约算法来解决配送路线中的车辆安排问题. 闫军等^[21]根据配送服务的特性, 设计一种将聚类处理与蚁群算法相结合的混合算法来解决两指标的 PDPTW 问题. 陈凯等^[22]通过引入移除-修复算子, 设立带评分的领域搜索策略, 在保留算法核心搜索机制的基础上, 设计了一种改进的离散灰狼优化算法, 旨在解决带时间窗约束的同时取送货车辆路径问题. 刘建胜等^[23]对头脑风暴算法进行改进, 提出自适应头脑风暴算法, 用于求解车辆使用成本和车辆行驶距离成本最少的路径优化模型.

综合相关研究来看, 国内外学者对于众包订单分配与路径规划问题, 提出的相关模型和算法基本成熟, 但实际众包配送服务中, 现有研究仍存在一些尚未解决的弊端. 首先, 现有研究主要集中分析不同车辆的载重、速度、成本等方面的差异, 以及单一服务类型配送员的不同等级、配送能力等方面的差异, 鲜少考虑在实际众包配送过程中, 不同服务类型的骑手对于配送同类或者不同类型订单在许多方面具有显著差异. 其次现有研究主要从平台、顾客或者骑手的角度出发, 考虑最小化配送平台的总成本、最大化顾客的满意度或者最大化骑手收益等目标, 而将商家和顾客的满意度同时考虑在内的较少. 因此, 与以往的研究不同, 本文研究的是具有多服务类型骑手的众包取送货车辆路径问题, 从众包平台角度出发, 兼顾商家和顾客的满意度, 旨在提升众包平台的配送效率以及降低平台的配

送成本.

2 问题分析及模型

2.1 问题描述

本文研究的是众包平台派单模式下订单分配与路径优化的实际场景, 即对于某一时刻产生的订单, 众包平台如何把这些不同类型的订单分配给不同服务类型的骑手, 并进行合理的配送路线规划, 以最小化平台总配送成本.

文章构建的具体场景为: 在某一区域内某众包平台拥有多个外卖、快递订单和多个可供分配的众包骑手, 骑手分为单一型骑手(单一外卖型、单一快递型)和全能型两种服务类型的骑手. 顾客在手机 APP 上提交外卖或者快递订单, 对于某一时刻的所有订单, 众包平台会将订单信息放入订单池, 通过算法处理完成对骑手的订单分配与路径规划工作, 将优化后的订单配送路径信息派发给骑手. 骑手分散在各个不同的地点, 具有不同的配送起点位置, 且骑手在完成订单任务后不用返回起点.

2.2 基本假设及参数说明

2.2.1 基本假设

- 1) 各个骑手的位置, 各个顾客与商家的位置、时间窗、最大容忍时间、费用系数均已知.
- 2) 同服务类型的骑手, 各项参数均相同.
- 3) 平台内众包骑手的数目远远满足平台的配送需求.
- 4) 骑手的实际配送时间仅考虑在配送路途中的行驶时间, 在取送货点的等待与服务时间均不计入骑手的配送时间.

2.2.2 符号说明

1) 集合

$N: N = P \cup D = \{1, 2, \dots, n, n+1, \dots, 2n\}$.

P : 商家(取货点)集合, $P = \{1, 2, \dots, n\}$.

D : 顾客(送货点)集合, $D = \{n+1, n+2, \dots, 2n\}$.

R : 众包骑手集合, $R = \{1, 2, \dots, k\}$, 分为 3 种不同服务类型的骑手.

2) 参数

k^0 : 众包骑手起始位置.

k^- : 众包骑手完成所有配送任务后不用返回起点, 假设众包骑手虚拟终点位置.

$[e_i, l_i]$: 节点*i*的期望服务时间窗, $\forall i \in N$.

e_i : 节点*i*的期望最早到达时间, $\forall i \in N$.

l_i : 节点*i*的期望最晚到达时间, $\forall i \in N$.

L_i : 节点*i*最大容忍时间, $\forall i \in N$.

d_{ij} : 从节点*i*到节点*j*的配送距离, $\forall i, j \in N$ 且*i* ≠ *j*.

T_i^k : 众包骑手从自己起点位置出发到达节点*i*的时间, $\forall i \in N, \forall k \in R$.

A_i^k : 商家和顾客对于众包骑手到达取送货节点的时间满意度, $\forall i \in N, \forall k \in R$.

P_i^k : 时间惩罚函数, $\forall i \in N, \forall k \in R$.

c_1 : 众包骑手单位提前到达时间参数.

c_2 : 在期望的超时范围内, 众包骑手单位超时时间参数.

c_3 : 到达时间超过可容忍的超时范围, 众包骑手额外单位超时时间参数.

c_4 : 众包骑手单位配送路径参数.

c_5 : 众包骑手单位人员运营参数.

v : 众包骑手的平均行驶速度.

3) 决策变量

$x_{i,j}^k$: 0-1 变量, 当骑手*k*经过节点*i*和节点*j*时为 1, 否则为 0, $\forall i, j \in N$ 且*i* ≠ *j*, $\forall k \in R$.

2.3 商家、顾客满意度成本

本文考虑的是商家和顾客对于骑手到达取货点和送货点的时间满意度的刻画. 现有研究大多使用软时间窗下的连续线性时间满意度函数来衡量客户的满意度变化情况^[24].

对于商家来说, 如果骑手到达的时间早于其期望的左时间窗, 这时候商家可能未准备好货, 那么商家的满意度随着骑手的到达时间线性上升; 如果骑手到达时间晚于其期望的右时间窗, 这个时候货物早已备好, 一些货物比如水果生鲜的质量会下降, 此时商家的满意度也会随着骑手的到达时间线性下降; 如果骑手到达商家的时间在其期望的时间窗内, 则商家的满意度为 1.

同理, 对于顾客来说, 当骑手到达的时间在其期望的时间窗之外, 那么顾客的满意度随着骑手到达时间与期望时间差值的增大呈线性变化, 如果骑手到达顾客点的时间在其期望的时间窗内, 则顾客的满意度为 1. 因此商家和顾客对于取送货时间的满意度, 可以看作同一个函数, 满意度成本函数和图像分别如式 (1) 和图 1 所示.

$$A_i^k = \begin{cases} \frac{T_i}{e_i}, & 0 < T_i < e_i \\ 1, & e_i \leq T_i \leq l_i \\ \frac{L_i - T_i}{L_i - l_i}, & l_i < T_i < L_i \\ 0, & T_i \geq L_i \end{cases} \quad (1)$$

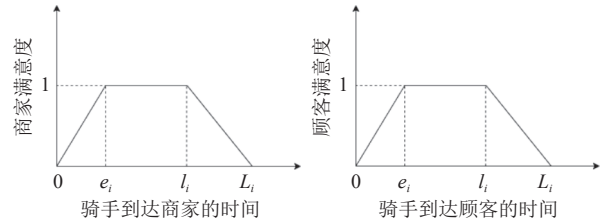


图 1 考虑模糊时间窗的商家和顾客满意度

客户满意度和交付成本是不同维度的, 因此需要规范, 将客户满意度函数转化为惩罚函数. 商家和顾客满意度最大可以转换成时间惩罚成本最小^[24], 转化后时间惩罚成本的函数图像如图 2 所示.

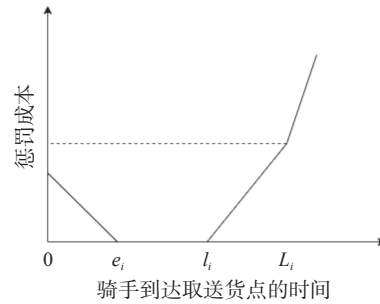


图 2 客户满意度转化为惩罚成本

对应的惩罚函数表示如下:

$$P_i^k = \begin{cases} c_1(e_i - T_i), & 0 < T_i < e_i \\ 0, & e_i \leq T_i \leq l_i \\ c_2(T_i - l_i), & l_i < T_i < L_i \\ c_2(L_i - l_i) + c_3(T_i - L_i), & T_i \geq L_i \end{cases} \quad (2)$$

在商家和顾客所期望的时间段 $[e_i, l_i]$ 内, 他们的满意度这时是最大的, 惩罚成本为 0; 在时间段 $[0, e_i]$ 内, 时间惩罚成本随时间向后推移而减少; 对于超时时间, 在时间段 (l_i, L_i) 内, 商家和顾客一般容易接受, 惩罚成本随着时间的向后推移逐渐增大; 在时间段 $[L_i, +\infty)$, 意味着超时时间超过最大容忍时间, 单位超时成本会比原来增大, 惩罚成本也随着时间的向后推移逐渐增大.

惩罚成本数学表达式为:

$$\sum_{i \in N} \sum_{k \in R} \{c_1 \max(e_i - T_i^k, 0) + c_2 \max(T_i^k - l_i, 0) + c_3 \max(T_i^k - l_i - L_i, 0)\} \quad (3)$$

其中, $\sum_{i \in N} \sum_{k \in R} c_1 \max(e_i - T_i^k, 0)$ 表示骑手提前到达时间在 $[0, e_i)$, 提前时间系数为 c_1 ; $\sum_{i \in N} \sum_{k \in R} c_2 \max(T_i^k - l_i, 0)$ 表示超时时间在 (l_i, L_i) 以内, 超时系数为 c_2 ; $\sum_{i \in N} \sum_{k \in R} c_3 \max(T_i^k - l_i - L_i, 0)$ 表示超时时间超过了最大容忍时间 L_i , 超时系数变大为 c_3 .

2.4 众包平台总配送成本

2.4.1 配送路径成本

配送路径成本指的是骑手完成所有节点取送货任务的配送成本, 包括骑手从当前位置到路线中第 1 个节点的距离成本以及骑手配送路线中各个节点之间的距离成本.

数学表达式为:

$$\sum_{i \in N} \sum_{j \in N} \sum_{k \in R} c_4 (x_{k^0, i}^k d_{k^0, i}^k + x_{i, j}^k d_{i, j}^k) \quad (4)$$

其中, $\sum_{i \in N} \sum_{j \in N} \sum_{k \in R} x_{k^0, i}^k d_{k^0, i}^k$ 表示骑手从当前位置到配送路线第 1 个节点的距离; $\sum_{i \in N} \sum_{j \in N} \sum_{k \in R} x_{i, j}^k d_{i, j}^k$ 表示骑手配送线路中各节点之间的距离.

2.4.2 人员运营成本

人员运营成本指的是在骑手完成所有节点取送货任务后, 众包平台会给予相应的津贴, 从而有效吸引和激励众包骑手, 长远发展运营平台. 众包平台对于每个类型的骑手完成一次配送任务的运营成本是一样的.

数学表达式为:

$$\sum_{j \in N} \sum_{k \in R} c_5 x_{k^0, j}^k \quad (5)$$

其中, $\sum_{j \in N} \sum_{k \in R} c_5 x_{k^0, j}^k$ 表示完成所有配送任务所需的骑手数与单位人员运营成本的乘积.

2.5 数学模型构建

数学模型的目标函数如下所示:

$$\begin{aligned} \min Z = & \sum_{i \in N} \sum_{k \in R} \{c_1 \max(e_i - T_i^k, 0) \\ & + c_2 \max(T_i^k - l_i, 0) + c_3 \max(T_i^k - l_i - L_i, 0)\} \\ & + \sum_{i \in N} \sum_{j \in N} \sum_{k \in R} c_4 (x_{k^0, i}^k d_{k^0, i}^k + x_{i, j}^k d_{i, j}^k) + \sum_{j \in N} \sum_{k \in R} c_5 x_{k^0, j}^k \end{aligned} \quad (6)$$

其中, 目标函数为最小化平台总配送成本, 包含 3 个部分: 完成所有订单的时间惩罚成本、路径行驶成本以及人员运营成本.

2.6 约束条件

数学模型的约束条件如下所示:

$$\sum_{i \in P} \sum_{j \in D} x_{i, j}^k = 1, \forall k \in R \quad (7)$$

$$\sum_{j \in N} \sum_{k \in R} x_{k^0, j}^k \leq R \quad (8)$$

$$\sum_{i \in N \cup \{k^-\}} x_{k^0, i}^k = 1, \forall k \in R \quad (9)$$

$$\sum_{i \in N \cup \{k^0\}} x_{i, k^0}^k = 1, \forall k \in R \quad (10)$$

$$\sum_{i \in N} x_{k^0, i}^k \leq 1, \forall k \in R \quad (11)$$

$$\sum_{j \in N \cup \{k^0\}} x_{j, i}^k - \sum_{j \in N \cup \{k^-\}} x_{i, j}^k = 0, \forall i \in N, \forall k \in R \quad (12)$$

$$\sum_{j \in N} x_{i, j}^k - \sum_{j \in N} x_{j, i+n}^k = 0, \forall i \in P, \forall k \in R \quad (13)$$

$$T_{k^0}^k = 0, \forall k \in R \quad (14)$$

$$T_j^k = T_i^k + \frac{d_{ij}}{v}, \forall k \in R, \text{ 且 } x_{i, j}^k = 1, \forall i \in P, \forall j \in D \quad (15)$$

$$T_i^k < T_{i+n}^k, \forall i \in P, \forall k \in R \quad (16)$$

其中, 式 (7)–式 (13) 表示配送路径选择相关约束, 式 (7) 表示每个订单仅能由一名众包骑手进行配送; 式 (8) 表示进行配送的众包骑手不能超过骑手总数; 式 (9)、式 (10) 表示每个骑手在配送路径上存在起点和终点, 也就是每个骑手从其起始点开始配送, 一直到终点结束配送任务, 这里的终点是虚拟终点, 从最后一个节点到它的距离不计入骑手的配送路径; 式 (11) 表示不一定每个众包骑手都被分配到订单; 式 (12) 表示所有骑手在抵达一个节点之后, 必定会从这个点离开; 式 (13) 表示订单一定是成对的; 式 (14)–式 (16) 表示配送时间相关约束, 式 (14) 表示众包骑手从自己起始位置出发的时刻设置为 0; 式 (15) 表示众包骑手从 i 点到达 j 点的时间; 式 (16) 表示满足订单中任务点配送的优先级约束, 即先取货后送货.

3 求解算法设计

3.1 基本模拟退火算法

模拟退火算法流程图见图 3, 具体步骤如下.

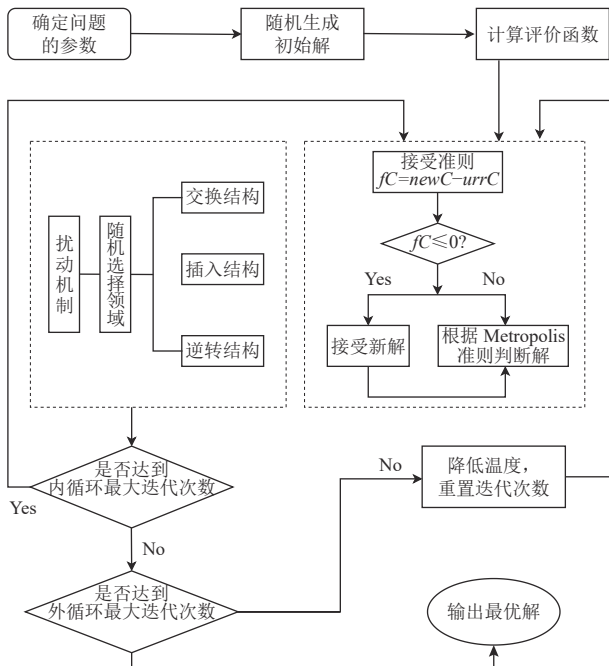


图 3 模拟退火算法流程图

步骤 1. 设计初始温度 $T = T_0$, 根据订单类型和骑手服务类型的多样性, 随机产生一个初始解 $currS$, 执行解码操作并计算出对应的评价函数 $currC$.

步骤 2. 对当前解 $currS$ 通过扰动机制产生一个新解 $newS$, 并计算出对应的评价函数 $newC$. 扰动机制为随机选择邻域操作, 由预先设定的概率参数来动态选择执行的 3 种领域结构操作: 交换结构、逆转结构和插入结构, 从而改变当前解的配置. 交换结构通过随机交换路线中的两个订单的位置, 逆转结构反转选择的子路线顺序, 插入结构将随机选择的订单插入到另一位置.

步骤 3. 接受准则基于新解与当前解的目标函数值差异以及当前温度来决定是否接受新解. 在当前温度 T 下, 计算初始解 $currS$ 与新解 $newS$ 的评价函数之差 fC ($fC = newC - currC$), 如果 $fC < 0$, 说明新产生的解比当前解更优秀, 算法始终接受新解; 反之, 则要根据接受概率 P 来判断是否接受新解. 随机生成 0-1 之间的概率数 $rand$, 比较 $rand$ 与 P 的大小, 如果 $rand < P$, 则接受新解; 反之, 就保持当前解 $currS$ 与评价函数 $currC$.

概率 P 的计算公式如下:

$$P = \exp\left(-\frac{\delta}{T}\right) = \exp\left(-\frac{newS - currS}{currS \times T}\right) \quad (17)$$

其中, δ 表示当前解与新解的成本变化百分比, T 表示当前温度.

步骤 4. 比较当前内循环次数 $inIter$ 与当前温度下的最大内循环次数 $MaxInIter$, 若 $inIter \leq MaxInIter$, 则重复扰动机制和接受准则过程; 否则, 根据冷却因子 α 来降低当前温度, 记录当前全局最优解 $bestC$ 并跳出内循环.

步骤 5. 内层循环完成后, 更新当前温度, 比较当前外层循环次数 $outIter$ 与最大外层循环次数 $MaxOutIter$, 若 $outIter \leq MaxOutIter$, 则进入下一次迭代, 重复内层循环扰动机制和接受准则等过程, 否则终止算法, 输出当前全局最优解.

3.2 基本遗传算法

遗传算法具体步骤如下.

步骤 1. 由于众包订单配送在路线的规划中, 必须兼顾订单的配对和配送优先性, 所以本文采用整数编码方式进行染色体编码. 考虑到订单类型的多样性 (外卖订单、快递订单) 以及骑手的不同服务类别 (单一外卖型、单一快递型和全能型), 将初始种群的生成分为如下几步: 首先生成局部的初始种群, 包括单一外卖型初始编码序列 $Chrom1$ 、单一快递型初始编码序列 $Chrom2$ 和全能型初始编码序列 $Chrom3$; 接着将 3 种不同类型的编码序列综合到一起, 形成一个包含多种订单服务请求类型的综合初始种群; 然后对每个个体的编码序列进行检查, 识别并删除其中的重复节点, 得到最终的初始种群.

步骤 2. 解码需要先提取染色体中每个骑手的订单配送路径信息, 来识别所有分割点的位置. 第 1 个节点 0 以前的所有节点都表示编号为 1 的骑手的配送任务路径; 第 $m-1$ 个节点和第 m 个节点之间的所有节点表示编号为 m 的骑手的配送任务路径; 如果两个 0 节点是相邻的, 这就意味着对应编号的骑手没有被分配到任何订单, 重复直至提取完染色体中的所有配送任务路径信息 (不含空路径).

步骤 3. 根据平台的总配送成本计算适应度函数.

步骤 4. 采用精英选择策略将预先设定的一定数量的较优个体从种群中选出, 直接进入下一代, 对于剩余个体采用随机遍历抽样策略挑选出合适个体填充子代种群.

步骤 5. 考虑到订单与骑手的类型是否匹配, 本文采用顺序交叉方式, 以一定的交叉概率对选择的染色体进行交叉操作进行交叉操作. 随机选取两对父代染色体, 从中提取第 1 个骑手的基因片段, 将其交叉放在

父代染色体的最前端,并删除染色体中的重复节点。

步骤6. 考虑到订单与骑手的类型是否匹配,采用随机两点变异进行变异操作. 从父代染色体中提取全能型骑手的基因片段,在全能型基因片段上随机选择两个不同的取送货节点对基因进行置换。

3.3 混合算法设计

作为启发式算法,遗传算法、模拟退火算法都存在收敛速度较慢、搜索能力不强、运行时间较长等缺点^[3],因此本文采取遗传算法与大领域搜索算法相结合的方式,先采用遗传算法进行全局搜索,然后再用大领域搜索算法进行局部搜索,来求解众包配送过程中订单分配和路线规划组合优化问题. 局部搜索操作的具体步骤如下。

3.3.1 移除操作

在ALNS中, $Chrom$ 表示当前种群, $visit$ 表示初始被选定移出的订单取货节点, $toremove$ 表示设定的总移除算子数量, in 表示执行移除操作后的剩余节点集合,在执行移除操作前集合 $in = \{1, 2, \dots, n, n+1, \dots, 2n\}$,移除操作的过程分为以下几个步骤。

首先,从所有取送货节点的集合 in 中随机选出一个取货节点,并寻找其对应的送货节点,将取货节点和送货节点放入 $removed$ 中,并从 in 中移除这两个节点;其次,将剩下的 $toremove - 1$ 个取送货节点按照如下规则移出:先从 $removed$ 内随机选取一个取货节点,计算其与 in 内所有节点的相关性;接着根据相关性高低将 in 中节点进行依次排序,选取一个相关性最高的取货节点 vr ,找到其对应的送货节点 vrr ;然后将 vr 与 vrr 依次放入 $removed$ 中,并从 in 中移除这两个取送货节点;重复该操作,直到 $removed$ 中节点数量与预先设定的移除点数量 $toremove$ 相同,就停止操作。

相关性 R_{ij} 的计算公式如下:

$$R_{ij} = 1 / (V_{ij} + D_{ij} + T_{ij} + O_{ij} + K_{ij}), \forall i \in removed, \forall j \in in \quad (18)$$

$$V_{ij} = \sum_{m \in N} x_{im}^k \sum_{m \in N} x_{mj}^k, \forall i \in removed, \forall j \in in, k \in R \quad (19)$$

$$D_{ij} = d_{ij} / \max\{d_{ij}\}, \forall i \in removed, \forall j \in in \quad (20)$$

$$T_{ij} = |e_i - e_j| / \max\{|e_i - e_j|\}, \forall i \in removed, \forall j \in in \quad (21)$$

$$O_{ij} = \begin{cases} 0.1, & \text{如果 } i \text{ 与 } j \text{ 订单类型相同} \\ 0.7, & \text{如果 } i \text{ 与 } j \text{ 订单类型不相同} \end{cases} \quad (22)$$

其中, D_{ij} 和 T_{ij} 的取值范围均为 $[0, 1]$, V_{ij} 表示当节点 i 和节点 j 在同一个配送路径中则为1,否则为0; D_{ij} 表示两个节点之间欧氏距离越近,相关性就越高; T_{ij} 表示不同节点期望的骑手送达时间窗下限差值的绝对值越小,节点之间相关性就越高; O_{ij} 表示如果节点 i 和节点 j 的类型一样,则节点之间相关性就越高; K_{ij} 表示如果节点 j 是取货节点,则节点之间相关性就直接设置为0。

3.3.2 修复操作

修复操作是将被移除的取送货节点对重新插回到解中。

首先,从 $removed$ 中随机选择一个取货节点 rv ,从当代染色体中提取每个骑手的配送路径,根据取货节点 rv 在第 k 个配送路径中的每个插入位置,依次遍历每个插入位置,当取货节点 rv 插入到第 k 个骑手路径中的某个位置时,如果路径中每个节点 j 满足以下约束条件,则该位置为取货节点 rv 的可行插入位置,否则第 k 个配送路径不予考虑:(1)第 k 个配送路径对应的骑手是否可以配送取货节点 rv ,如果不满足条件,直接跳过 k 个配送路径;(2)满足配送服务类型要求后,将取货节点 rv 和其送货节点 rvv 按顺序插入,插入后此路径中所有节点都满足期望时间窗约束.其次,采用最近插入启发式算法计算所有合理插入点的距离增量,通过对比所有距离增量的大小,将最小距离增量元素对应的插入位置找出来,这就是该取送货节点的最优插入位置.然后,对比 $removed$ 中每个取货点的最优插入位置的距离增量,采用最远插入启发式算法,找出距离增量最大的取货点并插入到对应路径中,随后更新该条路径并删除 $removed$ 中该取送货节点对.如果出现一个取货节点在路径中没有可行的插入位置,则停止修复操作,当前种群就为未进行移除-修复前的种群.最后,每插回一个取送货节点则重新寻找 $removed$ 中剩余取送货节点对的最优插入位置,循环上述操作直到 $removed$ 为空,并更新配送路径信息。

3.4 其他操作

3.4.1 删除重复的染色体

由于变异概率较低以及算法的迭代更新,新生成的个体与其附带个体之间存在的差异可能非常小,容易产生重复个体.设计有效的机制,去除重复个体,维持种群多样性,增强算法的全局搜索能力。

3.4.2 补足染色体种群

删除重复的染色体后,种群数量会相对变少,为了

维持种群的多样性,提升算法的求解速度与收敛性,随机生成新的个体来补足种群规模。

3.4.3 算法终止条件及流程图

当算法达到预先设定迭代次数时,就停止运行算法.混合算法的流程图如图4所示。

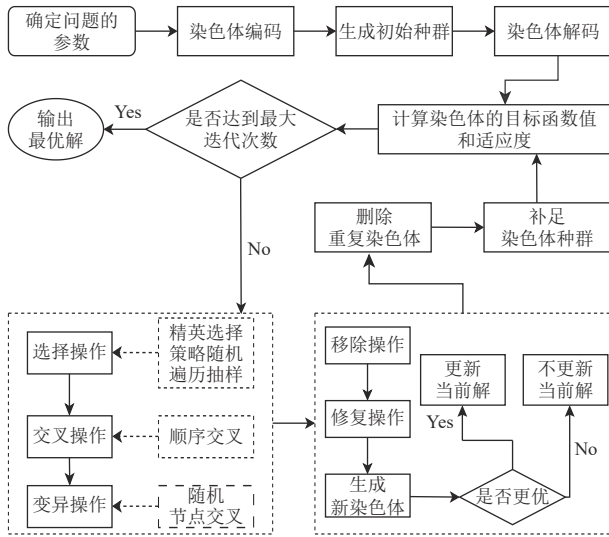


图4 混合算法流程图

4 仿真实验与结果分析

本文利用 Matlab R2020b 进行编程求解,所有数据均在 11th Gen Intel(R) Core(TM) i7-1195G7 @ 2.90 GHz 配置的计算机上进行仿真实验.具体如下。

4.1 算例选取

选取某众包平台在某一时刻生成的 30 个订单数据,随机将数据中的节点设定为外卖节点(标记为 0)和快递节点(标记为 1),将数据转换成本文所需的订单算例.为了确保众包骑手的数量能够满足该地区的订单配送任务,假设众包平台可以参加配送任务的骑手有 15 人,在该平台设定的配送区域内随机生成 15 个众包骑手的位置点信息,并在此基础上生成 15 个骑手的各个订单类型(单一外卖型骑手标记为 0,单一快递型骑手标记为 1,全能型骑手标记为 2).算例数据如表 1 和表 2 所示。

4.2 参数

有关参数设置如表 3 所示。

4.3 模拟退火算法求解结果

如图 5,通过模拟退火算法可以求解得出,该模型的目标函数值为 220.12 元,配送路线总长为 51.89 km,

超时时间总计为 46.71 min,使用的骑手数为 9 个,具体配送路线如表 4 所示。

表 1 骑手算例数据

编号	坐标	骑手类型	编号	坐标	骑手类型
1	(50, 85)	0	9	(35, 66)	0
2	(56, 24)	2	10	(60, 67)	0
3	(18, 14)	1	11	(50, 47)	1
4	(20, 11)	0	12	(33, 15)	2
5	(10, 8)	0	13	(89, 83)	0
6	(38, 40)	0	14	(66, 10)	1
7	(82, 43)	1	15	(71, 46)	2
8	(37, 58)	2			

表 2 订单算例数据

编号	坐标	时间窗	L_i	类型	编号	坐标	时间窗	L_i	类型
1	(2.62, 7.43)	[0, 15]	25	0	31	(2.01, 8.07)	[0, 38]	48	0
2	(1.94, 7.34)	[0, 15]	25	1	32	(2.1, 6.69)	[0, 32]	42	1
3	(2.69, 7.11)	[0, 15]	25	0	33	(1.92, 8.02)	[0, 37]	47	0
4	(1.97, 7.41)	[0, 15]	25	1	34	(0.51, 6.71)	[0, 40]	50	1
5	(2.25, 7.24)	[0, 15]	25	1	35	(2.33, 7.48)	[0, 31]	41	1
6	(1.94, 7.4)	[0, 15]	25	0	36	(2.66, 7.41)	[0, 31]	45	0
7	(2.62, 7.43)	[1, 16]	26	0	37	(3.99, 8.33)	[1, 31]	48	0
8	(3.4, 6.84)	[1, 16]	26	1	38	(1.07, 6.72)	[1, 39]	49	1
9	(1.94, 6.88)	[1, 16]	26	0	39	(2.25, 7.34)	[1, 38]	48	0
10	(2.53, 6.98)	[1, 16]	26	0	40	(0.43, 7.28)	[1, 44]	54	0
11	(2.99, 7.43)	[1, 16]	26	0	41	(3.75, 8.58)	[1, 38]	48	0
12	(2.75, 7.97)	[1, 16]	26	1	42	(3.47, 8.45)	[1, 38]	48	1
13	(2.68, 7.09)	[2, 17]	27	0	43	(2.25, 7.33)	[2, 34]	44	0
14	(2.64, 7.03)	[2, 17]	27	1	44	(1.27, 5.97)	[2, 43]	53	1
15	(1.94, 6.84)	[2, 17]	27	0	45	(1.53, 5.88)	[2, 46]	56	0
16	(1.94, 6.84)	[2, 17]	27	0	46	(2.33, 8.39)	[2, 50]	60	0
17	(2.88, 7.28)	[3, 18]	28	1	47	(4.45, 8.32)	[3, 45]	55	1
18	(1.94, 6.88)	[3, 18]	28	1	48	(1.5, 7.65)	[3, 40]	50	1
19	(2.76, 7.12)	[3, 18]	28	0	49	(3.77, 7.35)	[3, 37]	47	0
20	(1.95, 7.33)	[3, 18]	28	1	50	(1.76, 6.93)	[3, 38]	48	1
21	(1.93, 7.35)	[4, 19]	29	0	51	(1.88, 7.48)	[4, 36]	46	0
22	(2.68, 7.09)	[4, 19]	29	0	52	(1.9, 7.04)	[4, 41]	51	0
23	(3.05, 7.44)	[4, 19]	29	0	53	(2.8, 7.33)	[4, 33]	43	0
24	(2.76, 7.12)	[4, 19]	29	1	54	(1.98, 6.79)	[4, 34]	44	1
25	(1.94, 7.34)	[4, 19]	29	0	55	(3.6, 9.13)	[4, 47]	57	0
26	(3, 7.47)	[5, 20]	30	0	56	(4.81, 7.51)	[5, 37]	47	0
27	(2.26, 7.24)	[5, 20]	30	0	57	(1.62, 7.99)	[5, 39]	49	0
28	(3.03, 7.88)	[5, 20]	30	1	58	(2.11, 6.69)	[5, 53]	63	1
29	(1.95, 7.34)	[5, 20]	30	0	59	(1.58, 6.71)	[5, 41]	51	0
30	(2.68, 7.09)	[5, 20]	30	1	60	(1.6, 8)	[5, 42]	52	1

4.4 遗传算法求解结果

如图 6,通过遗传算法可以求解得出,该模型的目标函数值为 215.402 元,配送路线总长为 47.35 km,超时时间总计为 38.25 min,使用的骑手数为 10 个,具体配送路线如表 4 所示。

4.5 混合算法求解结果

如图 7,通过混合算法可以求解得出,该模型的目的

标函数值是 181.42 元, 配送路线总长为 49.23 km, 超时时间总计为 6.31 min, 使用的骑手数为 8 个, 配送路线如表 4 所示.

表 3 参数设置

参数名称	名称	设置数值
种群大小	<i>NIND</i>	100
迭代次数	<i>MAXGEN</i>	300
交叉概率	<i>Pc</i>	0.9
变异概率	<i>Pm</i>	0.06
代沟	<i>GGAP</i>	0.9
染色体长度	<i>N</i>	74
外循环最大迭代次数	<i>MaxOutIter</i>	800
内循环最大迭代次数	<i>MaxInIter</i>	500
初始温度	<i>T0</i>	100
冷却因子	<i>alpha</i>	0.99
配送速度	<i>v</i>	20 km/h
单位提前时间成本	<i>c₁</i>	0.04 元/min
单位超时时间成本	<i>c₂</i>	0.05 元/min
单位额外超时时间成本	<i>c₃</i>	1 元/min
单位配送路径成本	<i>c₄</i>	2 元/min
单位人员运营成本	<i>c₅</i>	10 元/人

4.6 结果对比分析

从图 5-图 7 的结果对比可以得出, 在基于算例的求解过程中, 混合算法展现出了较优的求解性能, 更适用于解决具有异质性的众包配送路径规划问题. 具体来看, 模拟退火算法和遗传算法分别在第 320 代、第 221 代收敛, 而混合算法在第 24 代收敛. 在算法的中期阶段, GA 和 SA 就已陷入局部最优, 而混合搜索算法具有一定的局部搜索能力, 在逐渐接近问题的最优解.

通过分析进一步表明, 局部搜索操作的加入, 使得每次遗传迭代后的染色体经过移除与修复操作, 进一步得到了优化. 移除操作有选择地从当前解中删除一些元素, 使得算法能够离开当前解所在的局部最优区域, 探索新的可能性空间, 有效避免 GA、SA 在迭代过程中出现早熟收敛以及依赖初始解的情况. 修复操作在遵循一系列约束条件的情况下, 将受损的染色体进行修复, 生成改良后的新染色体, 维持了种群的多样性. 这种新产生的染色体显著提高了质量, 从而有效增强了算法的收敛速度, 以及全局搜索和局部搜索的效率.

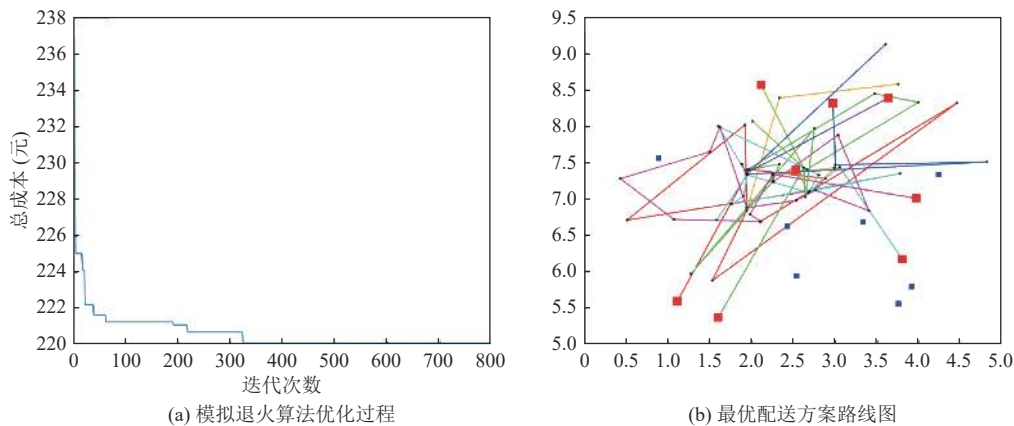


图 5 模拟退火算法迭代次数与配送路径图

表 4 最优路径方案 (骑手编号: 配送路径)

模拟退火算法	遗传算法	混合算法
1: 2-4-17-3-34-33-15-45-47-32	1: 6-36-17-11-47-41-1-31-4-34	1: 9-2-6-12-32-36-10-7-40-39-42-37
4: 13-11-43-16-46-41	3: 18-48	4: 17-47
6: 1-31	4: 12-42	5: 16-13-43-11-1-5-31-46-35-41
7: 7-14-12-44-5-35-6-42-37-36	5: 7-37	6: 14-4-44-34
8: 23-30-20-60-53-22-19-49-50-52-29-59	6: 8-10-14-38-40-44-3-33	9: 25-21-26-23-24-22-53-56-54-51-52-55
9: 26-56-25-55	7: 15-2-32-45-9-39	11: 20-19-50-28-49-58
11: 21-51	9: 19-22-49-28-58-52	12: 8-38-15-29-59-45
12: 9-27-28-8-39-57-38-58	10: 23-27-29-30-60-53-20-50-24-57-54-59	13: 3-30-27-33-18-57-60-48
13: 24-10-18-40-48-54	11: 25-55	
	12: 21-51-13-16-43-5-35-26-46-56	

由表 5 中算法优化结果对比可知, 本文的混合算法主要有以下优点.

(1) 模拟退火算法、遗传算法与混合算法的目标函数值分别为 220.12 元、215.40 元、181.42 元, 混合

算法的目标函数相比于模拟退火算法和遗传算法分别下降了 38.7 元、33.98 元, 目标函数值质量分别提升了 17.59%、15.78%。

(2) 在时间惩罚成本方面, 模拟退火算法、遗传算法与混合算法中骑手提前到达商家或者顾客的总时间分别为 1.42 min、0.54 min、0, 到达商家和顾客的时间超过其期望的右时间窗但低于最大容忍时间的总超

时时间分别为 41.89 min、34.45 min、4.42 min, 到达商家和顾客的时间超过其最大容忍时间分为 4.82 min、3.27 min、1.89 min; 模拟退火算法、遗传算法与混合算法的骑手总时间惩罚成本分别为 26.33 元、20.70 元、2.97 元, 混合算法相比较模拟退火算法与遗传算法分别下降了 88.72%、85.65%, 这同时也代表着商家和顾客的时间满意度在增加。

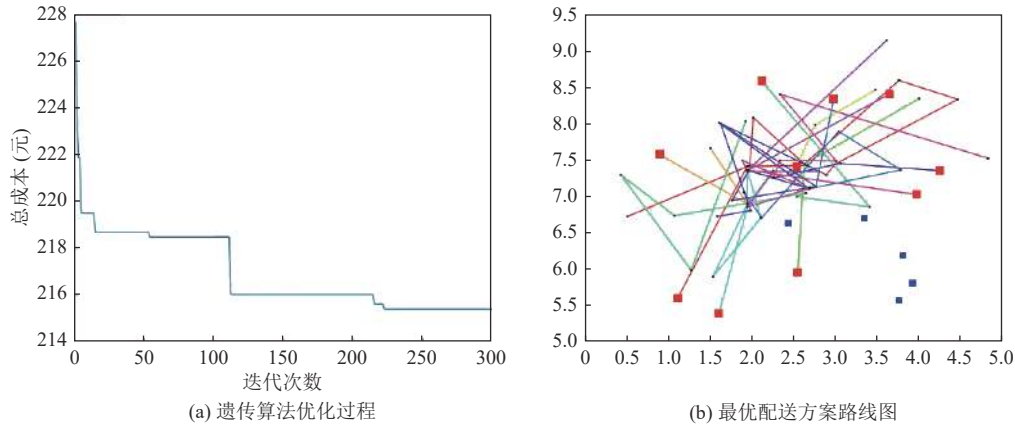


图 6 遗传算法迭代次数与配送路径图

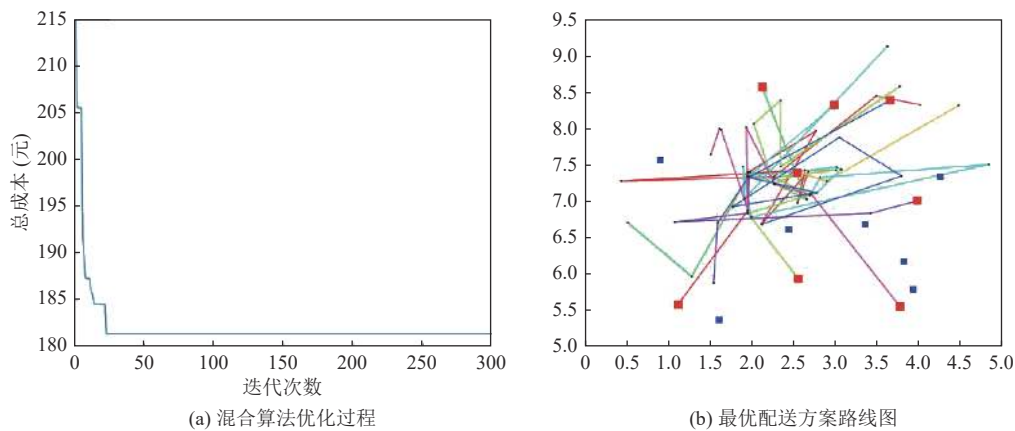


图 7 混合算法迭代次数与配送路径图

表 5 算法优化结果对比

名称	模拟退火算法	遗传算法	混合算法
目标函数(元)	220.12	215.40	181.42
时间惩罚成本(元)	26.33	20.70	2.97
路径行驶成本(元)	103.79	94.70	98.45
人员运营成本(元)	90	100	80
提前到达时间(min)	1.42	0.54	0
超过期望时间但不超过容忍时间(min)	41.89	34.45	4.42
超过容忍时间(min)	4.82	3.27	1.89
行驶路径长度(km)	51.89	47.35	49.23
所需人员个数	9	10	8

(3) 在路径行驶成本方面, 模拟退火算法和混合算法的总行驶距离分别为 51.89 km、49.23 km, 总路径行驶成本分别为 103.79 元、98.45 元, 相比较下混合算法总路径行驶成本下降了 5.14%。

(4) 在人员运营成本方面, 模拟退火算法所需骑手数为 9 个, 其中单一外卖型骑手有 4 个, 单一快递型骑手有 3 个, 全能型骑手有 2 个; 遗传算法所需骑手数为 10 个, 其中单一外卖型骑手有 6 个, 单一快递型骑手有 3 个, 全能型骑手有 1 个; 混合算法所需骑手数为 8 个, 其中单一外卖型骑手有 4 个, 单一快递型骑手有 2 个,

全能型骑手有2个。模拟退火算法、遗传算法与混合算法骑手总运营成本分别为90元、100元、80元, 相较于混合算法骑手总运营成本分别下降了10%、20%。

因此, 本文设计的混合算法可以有效减少骑手的总行路径驶距离, 以及降低骑手提前到达时间和超时时间, 也就是能有效提高顾客和商家的满意度。在保持较低路径行驶成本和较低惩罚时间成本(也就是较高商家和顾客满意度)的同时, 还可以减少众包骑手的使用数目, 提高众包平台人员利用率, 从而提高众包平台配送效率, 减少众包平台总配送成本。

5 结论

针对当前众包平台面临的订单类型多样性和配送骑手的单一型, 即外卖骑手仅负责外卖订单的配送, 快递骑手仅处理快递订单。本文考虑了多服务类型骑手的众包配送策略问题, 同时考虑到当前研究中较少同时关注商家和顾客满意度的情况, 构建基于模糊时间窗的商家和顾客对于骑手到达取送货点时间的满意度成本函数, 将商家和顾客的满意度转化为时间惩罚函数, 建立了以时间惩罚成本、路径行驶成本、人员运营成本最小化为目标的模型; 在算法求解方面, 考虑到传统启发式算法(如遗传算法、模拟退火算法)存在的不足以及待求解问题的复杂性, 设计了一种遗传算法与局部搜索算法相结合的混合算法进行求解, 运用算例, 采用不同算法进行仿真实验以及求解结果对比, 验证了本文构建的模型以及设计的求解算法的可行性和有效性。虽然本文建立的众包配送模型有一定的实践价值, 但是在建模过程中没有考虑到一些现实因素, 存在以下几点不足, 可作为下一步研究方向。

(1) 对于商家和顾客的满意度, 本文只考虑了骑手到达取送货点的时间对商家和顾客满意度的影响, 在现实生活中, 商家和顾客的满意度还会受到诸多因素的影响, 例如订单的配送费用、订单的处理速度、骑手的服务态度、货物的配送质量、平台的使用情况等, 需要多维度考虑。

(2) 本文考虑的是静态场景下的众包配送问题, 而在实际生活中, 外卖订单和快递订单具有动态性和高频性, 需要考虑订单动态产生的情况。

参考文献

1 赵建有, 李玥, 田浩, 等. 众包配送研究综述. 交通运输工程

- 学报, 2023, 23(5): 62–84. [doi: 10.19818/j.cnki.1671-1637.2023.05.004]
- 2 杜子超, 卢福强, 王素欣, 等. 众包物流配送车辆调度模型及优化. 东北大学学报(自然科学版), 2021, 42(8): 1210–1216. [doi: 10.12068/j.issn.1005-3026.2021.08.021]
- 3 李雪妍. 派单模式下考虑“取餐+送餐”的众包外卖配送优化研究[硕士学位论文]. 北京: 北京交通大学, 2019. [doi: 10.26944/d.cnki.gbfnj.2019.001020]
- 4 Arslan AM, Agatz N, Kroon L, *et al.* Crowdsourced delivery—A dynamic pickup and delivery problem with Ad Hoc drivers. *Transportation Science*, 2019, 53(1): 222–235. [doi: 10.1287/trsc.2017.0803]
- 5 王爽, 陈淮莉. 众包模式参与下“最后一公里”协同配送研究. *计算机应用与软件*, 2022, 39(10): 315–321. [doi: 10.3969/j.issn.1000-386x.2022.10.046]
- 6 邓娜, 张建军. O2O 外卖订单配送任务分配模式研究. *上海管理科学*, 2018, 40(1): 63–66. [doi: 10.3969/j.issn.1005-9679.2018.01.012]
- 7 戴韬, 沈静. 基于众包的外卖配送订单选择研究. *工业工程*, 2021, 24(2): 125–133. [doi: 10.3969/j.issn.1007-7375.2021.02.016]
- 8 冯爱兰, 周映雪, 龚艳茹, 等. 抢派结合模式下外卖配送问题研究. *控制与决策*, 1–8. <https://doi.org/10.13195/j.kzyjc.2022.1420>. (2023-10-08)[2024-03-06].
- 9 刘春玲, 王俊峰, 黎继子, 等. 众包模式下冷链物流配送模型的仿真和优化分析. *计算机集成制造系统*, 2019, 25(10): 2666–2675. [doi: 10.13196/j.cims.2019.10.025]
- 10 罗明亮, 袁鹏程. 考虑客户满意度的车辆路径优化及其算法研究. *河南师范大学学报(自然科学版)*, 2024, 52(2): 51–61. [doi: 10.16366/j.cnki.1000-2367.2022.10.01.0002]
- 11 陈星光, 李心宇, 程鲁强. 考虑订单偏好的外卖众包配送优化策略研究. *中国管理科学*, 2024, 32(6): 86–97. [doi: 10.16381/j.cnki.issn1003-207x.2021.1262]
- 12 Ren T, Xu HB, Jin KN, *et al.* Optimisation of takeaway delivery routes considering the mutual satisfactions of merchants and customers. *Computers & Industrial Engineering*, 2021, 162: 107728. [doi: 10.1016/j.cie.2021.107728]
- 13 孙云霄, 马继东. 基于配送员等级的多目标众包配送路径优化研究. *中国新技术新产品*, 2023(10): 134–136. [doi: 10.13612/j.cnki.cntp.2023.10.012]
- 14 刘烁佳, 李学强. 基于启发式带时间窗的车辆路径规划问题求解. *计算机系统应用*, 2022, 31(11): 275–281. [doi: 10.15888/j.cnki.csa.008818]
- 15 赵佳欣, 雷斌, 王菀莹. 时变网络下多车型同时取送货车辆路径优化. *计算机工程与设计*, 2023, 44(10): 3096–3102. [doi:

- 10.16208/j.issn1000-7024.2023.10.028]
- 16 张力娅, 张锦, 肖斌. 考虑顾客优先级的多目标 O2O 外卖即时配送路径优化研究. 工业工程与管理, 2021, 26(2): 196–204. [doi: 10.19495/j.cnki.1007-5429.2021.02.024]
- 17 熊浩, 郭昊颖, 鄢慧丽, 等. 外卖配送路径多目标实时优化研究. 工业工程, 2023, 26(1): 98–107. [doi: 10.3969/j.issn.1007-7375.2023.01.011]
- 18 陈高华, 郗传松. 求解多目标车辆路径优化的改进蚁群算法研究. 机械设计与制造, 2023(9): 231–236. [doi: 10.19356/j.cnki.1001-3997.20230322.004]
- 19 Ibrahim MF, Nurhakiki FR, Utama DM, *et al.* Optimised genetic algorithm crossover and mutation stage for vehicle routing problem pick-up and delivery with time windows. IOP Conference Series Materials Science and Engineering, 2021, 1071: 012025. [doi: 10.1088/1757-899X/1071/1/012025]
- 20 田亮, 李亚东. 考虑时间惩罚成本的配送路线优化. 物流工程与管理, 2020, 42(4): 105–109. [doi: 10.3969/j.issn.1674-4993.2020.04.034]
- 21 闫军, 常乐, 王璐璐, 等. 带时间窗的同时取送货车辆路径问题求解算法. 工业工程, 2021, 24(5): 72–76. [doi: 10.3969/j.issn.1007-7375.2021.05.009]
- 22 陈凯, 邓志良, 龚毅光. 离散灰狼优化算法求解 VRPSPDTW 问题. 计算机系统应用, 2023, 32(11): 83–94. [doi: 10.15888/j.cnki.csa.009305]
- 23 刘建胜, 蔡祥, 黄纪绘, 等. 考虑同时取送和时间窗的车辆路径及求解算法. 计算机工程与应用, 2023, 59(16): 295–304. [doi: 10.3778/j.issn.1002-8331.2205-0405]
- 24 任腾, 陈玥, 向迎春, 等. 考虑客户满意度的低碳冷链车辆路径优化. 计算机集成制造系统, 2020, 26(4): 1108–1117. [doi: 10.13196/j.cims.2020.04.024]

(校对责编: 孙君艳)