

# 当日达模式下考虑无人机补货的卡车无人机协同配送问题\*

马云峰<sup>1,2</sup>, 欧阳立君<sup>1</sup>, 胡健<sup>1</sup>, 胡依娜<sup>1</sup>, 任亮<sup>1,2</sup>

(1. 武汉科技大学 管理学院; 2. 武汉科技大学 服务科学与工程研究中心, 武汉 430065)

**摘要:** 电商企业为满足客户对快递配送的时效性要求提出了当日达模式。为提高配送效率,降低配送成本,研究了无人机同时用于协同卡车配送和补货的问题。根据配送完成时间将1个完整配送周期划分成多次调度决策,每次调度决策的求解基于枚举,将问题分解为多个混合整数规划子问题,再采用改进的自适应大邻域搜索算法提高求解子问题的效率。实验结果表明:改进的自适应大邻域搜索算法能在不影响求解质量的同时节约求解时间;相较于无人机仅补货和无人机仅送货,无人机同时用于补货和送货能够有效提高配送效率、降低配送成本。

**关键词:** 当日达; 无人机补货; 卡车无人机协同配送

**中图分类号:** O229; U116; F252

**文献标志码:** A

**文章编号:** 1672-6693(2025)02-0014-15

2023年,中国邮政行业寄递业务量累计完成1 624.8亿件,同比增长16.8%。其中,快递业务量(不包含邮政集团包裹业务)累计完成1 320.7亿件,同比增长19.4%<sup>[1]</sup>。电子商务带来的快递量业务增长显著,但与传统的实体店购物相比,线上购物的最大缺点是缺乏即时满足感。为了满足客户的时效性需求,提高客户满意度,亚马逊、淘宝、京东等电商平台相继提供当日达模式,给顾客提供一种近似线下购物的即时体验。

Zhang等人<sup>[2]</sup>将当日达模式归为具有随机请求的动态车辆路径问题中的一类,客户订单动态下达,且所有订单均在配送中心发出,需在较短时间内送至客户手中。Azi等人<sup>[3]</sup>在对当日达模式的研究中,顾客动态下单并要求在当天的限定时间内配送易腐货物,车辆多次返回配送中心补货。Klapp等人<sup>[4-6]</sup>将当日达视为动态调度波问题,定义了1组等距时间波,在每个波次决策是否派遣车辆配送。Ulmer等人<sup>[7]</sup>通过1组取件点和1支自动驾驶车队进行配送,客户动态地将货物订购到取货点,自动驾驶车辆负责将货物从配送中心送至取货点。在配送中应用无人机能够提高效率,降低成本。现有关于当日达模式的研究中,已出现无人机的应用:Dayarian等人<sup>[8]</sup>提出将无人机补货应用于当日达,使用1架无人机给1辆卡车补货,再由卡车进行配送,卡车无需返回配送中心取货,提高了配送效率,但无人机仅用于补货;Chen等人<sup>[9]</sup>提出深度Q学习算法解决当日达问题,使用卡车和无人机进行配送,但卡车和无人机并不协同,各自执行配送任务。无人机的应用仍有广泛的研究空间,如何更好地应用无人机,使得当日达模式的配送成本更小、配送效率更高,具有十分重要的研究意义。

在无人机的应用研究中,Murry等人<sup>[10]</sup>提出的飞行伴随旅行商问题(the flying sidekick traveling salesman problem, FSTSP)考虑了1辆卡车和1架无人机协同的情况,且无人机允许多次飞行,有最远飞行距离限制,每次飞行仅可访问1个客户。无人机-旅行商问题(the traveling salesman problem with drone, TSP-D)为FSTSP的变种问题,Agatz等人<sup>[11]</sup>、Poikonen等人<sup>[12]</sup>为求解TSP-D提出了不同的精确算法,Yurek等人<sup>[13]</sup>、Ha等人<sup>[14]</sup>运用启发约束算法求解TSP-D。随着研究的不断深入,多无人机-旅行商问题(the traveling salesman problem with multiple drones, TSP-mD)也受到了广大学者的关注。Peng等人<sup>[15]</sup>、Moshref-Javadi等人<sup>[16]</sup>对TSP-mD的研究中,为简化问题,令卡车在无人机起飞后在原地等待无人机返航。Boysen等人<sup>[17]</sup>、Hu等人<sup>[18]</sup>则是通过限制卡车在特定路径上行驶来简化问题。Murray等人<sup>[19]</sup>对TSP-mD的研究没有做出上述限制,因模型能求解的规模很小,提出了启发约束算法用于求解问题。上述研究中无人机每次飞行仅可为1个客户提供服务,可多次访问的多无人机协同问题则更加复杂。Poikonen等人<sup>[20]</sup>提出了多次访问无人机路径问题(the  $k$ -multi-visit drone routing problem,  $k$ -MVDRP),在该问题中,相同位置起飞的所有无人机必须在相同位置降落,

\* 收稿日期:2024-07-28 修回日期:2024-09-28 网络出版时间:2025-04-29T16:38

资助项目:教育部人文社会科学项目(No. 19YJA630054);国家自然科学基金面上项目(No. 52475526)

第一作者简介:马云峰,男,教授,博士,研究方向为运筹优化、物流系统规划、管理定向分析,E-mail: mayunfeng@wust.edu.cn

网络出版地址:https://link.cnki.net/urlid/50.1165.n.20250429.1317.004

无人机起飞后卡车必须直接开向降落位置。Kitjacharoenchai 等人<sup>[21]</sup>提出了二级无人机车辆路径问题(the two-echelon vehicle routing problem with drones, 2EVRPD),卡车配备多个可多次访问的无人机,限制每个位置最多起飞或降落 1 次。Luo 等人<sup>[22]</sup>打破了上述限制,将可多次访问的多无人机协同配送问题一般化。上述文献中无人机的应用局限于卡车协同配送,未涉及到给卡车补货的情形。

Pina-Pardo<sup>[23]</sup>提出的带投放期和无人机补货的旅行商问题(the traveling salesman problem with release dates and drone resupply, TSPRD-DR)假定订单的下单时间在决策时已知,建立了 1 架无人机为 1 辆卡车补货的混合整数规划模型,在后续研究中 Pina-Pardo<sup>[24-25]</sup>从 2 个方面对该问题进行扩展:一是用 1 架无人机为多辆卡车补货;二是将动态问题建模为 Markov 决策过程,设计了基于重新优化的在线策略来用于对实时订单的决策。上述文献将无人机应用于给卡车补货,但配送过程仅由卡车独立完成。

现有研究中,无人机的应用集中于单一的送货过程或补货过程,未发现将无人机同时应用于送货和补货的文献。为降低配送成本、提高配送效率,进一步探索无人机在当日达模式下的应用,本文提出了考虑无人机补货的卡车无人机协同配送问题,将无人机既用于补货又用于送货,卡车搭载多个送货无人机负责送货,补货无人机负责将实时订单补给到卡车。该问题涉及到了动态订单处理、卡车与多无人机协同、路径规划,可以视为动态车辆路径问题的高阶变体,属于 NP-难问题。本文设计了基于枚举和混合整数规划模型的求解方案,其中混合整数规划模型以成本最小化为目标,为体现当日达的时效性要求,在成本中考虑了迟到的惩罚成本。为减少决策用时,本文还对自适应大邻域搜索算法的算子进行了针对性的改进,提出了改进的自适应大邻域搜索算法(adaptive large neighborhood search algorithm, ALNSA)。

## 1 问题描述

本文提出了当日达模式下考虑无人机补货的卡车无人机协同配送问题,优化的目标是成本最小化,成本包括配送过程中卡车和送货无人机的行驶成本、违背送达时间的惩罚成本、补货过程中补货无人机以及卡车前往补货点的行驶成本。

问题定义在图  $\xi=(V, A)$  中,其中  $V=v_0 \cup N \cup M$  表示相关位置的集合,  $v_0$  表示配送中心,  $N$  表示所有潜在顾客点的集合,  $M$  表示所有补货点的集合。订单由顾客在周期  $T$  内实时发出,在  $j$  处的订单  $o$  到达时间由  $\tau_j^o$  表示,服务保障时间为  $S$ ,按时送达的最迟时间  $\rho_j^o = \tau_j^o + S$ 。送达时间超过  $\rho_j^o$  的部分称为迟到时间,卡车配送的迟到时间和无人机配送的迟到时间分别由  $l_i^G, l_i^U$  表示,是惩罚成本的依据。

送货过程由 1 辆卡车搭载多架送货无人机协同完成。送货无人机单次行程可送多个包裹,完成 1 次行程后在卡车上换电之后即可再次出发。在同一卡车节点可同时进行多架无人机的起飞和降落操作。考虑到配送的客户点地形的多样性,选择灵活性较强的小型无人机进行送货。由于送货无人机和卡车的差异,平均行驶速度分别为  $v^{Uq}, v^G$ 。从位置  $i$  到  $j$  的路程分别为  $d_{ij}^U$  和  $d_{ij}^G$ ,时间分别为  $t_{ij}^{Uq}$  和  $t_{ij}^G$ 。由于电商订单的货物通常为小件,假设所有客户的订单具有相同的体积和质量,以表示送货无人机每次行程的最大容量。无人机单次行程的最大续航里程为  $C^{Uq}$ ,最大容量为  $Q^{Uq}$ 。由于配送过程在城市内,不涉及远距离的跨城配送,现有车辆足以保证 1 d 的行驶需求,所以不对卡车的续航里程做限制。卡车和无人机的送货上门时的服务时间分别为  $s^G, s^{Uq}$ 。

补货过程由补货无人机和卡车在约定的补货点  $m$  完成。补货指令发出之后,补货无人机装载货物从配送中心出发与卡车会面,将货物补给到卡车之后,返回配送中心等待下次指令,卡车补货结束后开始新一轮的配送。选用较大的无人机来执行补货任务不仅能一次补充更多货物,也能搭载更大电池提高续航里程。补货无人机的最大装载数量为  $Q^{Us}$ ,平均行驶速度为  $v^{Us}$ ,从位置  $i$  到  $j$  的路程与送货无人机一致为  $d_{ij}^U$ ,行驶时间为  $t_{ij}^{Us}$ 。考虑到一次补货数量较多,装卸货操作所需空间较大,设置补货只能在空间较大、可以执行补货的地点进行。补货点并不需要额外的硬件支持,只需要存在相对较大的空间即可,因此设置补货点不需要支付固定成本。假设设置的所有补货点都在补货无人机的续航范围内,所以不对补货无人机的续航里程做限制。

一个完整的周期包含以下事件:

- 1) 顾客全天都在下订单;
- 2) 第 1 次调度时卡车载着送货无人机和当前已下单的货物从配送中心出发,提供配送服务(图 1a);
- 3) 卡车和补货无人机接到配送中心下一次调度的补货指令,卡车和送货无人机在完成当前配送之后前往补货点(图 1b);

- 4) 补货无人机从配送中心出发前往指定补货点,在补货点将实时下达的新订单补充至卡车(图 1b);
- 5) 在补货结束之后,卡车搭载送货无人机执行新一轮的配送任务(图 1c);
- 6) 在周期快结束且配送完所有订单之后,卡车返回配送中心(图 1d)。

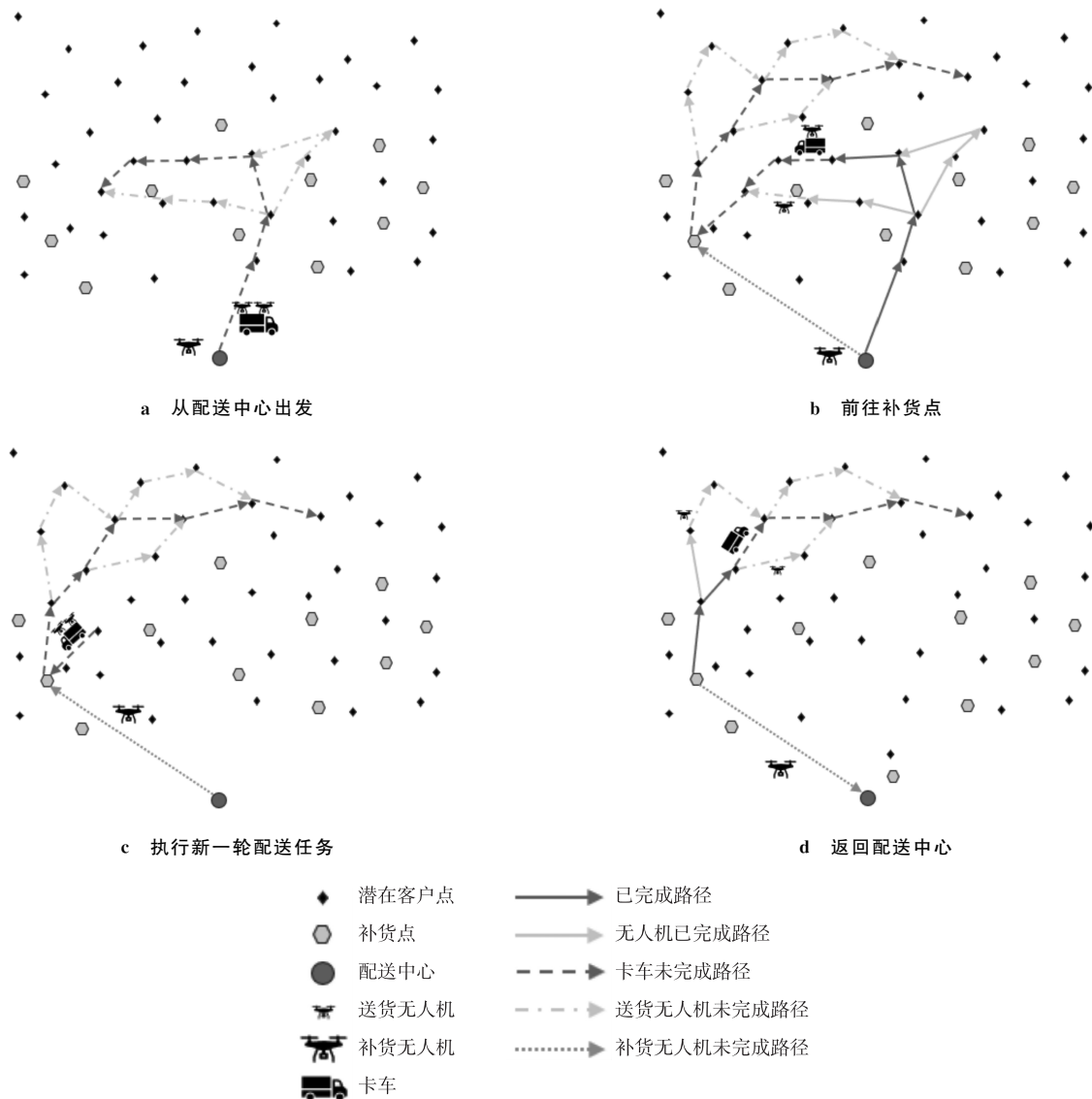


图 1 考虑无人机补货的车机协同配送示意图

Fig. 1 Truck-drone collaborative distribution with drone resupply

完成上述过程需要以下假设:

- 1) 允许延迟交货,无法按时配送的订单也将配送,以迟到时间作为惩罚成本的依据;
- 2) 在同一次配送过程中如果存在同一客户多次下单的情况,则将这些订单合并一次送达,以第 1 个订单的时间为惩罚成本的判断依据;
- 3) 如果卡车在收到补货指令之前完成了当前货物的配送,将会在配送的最后一个位置等待,直到配送中心发出新的指令;
- 4) 为了保证货物的安全性,不允许补货无人机先到达补货点等待卡车;
- 5) 在  $T-S$  后没有新订单产生,卡车允许在  $T$  之后返回配送中心。

## 2 求解方案设计

为求解上述问题,设计了基于枚举和混合整数规划模型的求解方案用于决策每次调度的补货点  $m^*$ 、调度时刻  $\mu^*$ 、计划的配送路径  $R^*$ 。

### 2.1 决策内容

送货过程决策的内容是配送路径  $R^*$ 。第 1 次调度不涉及补货过程,卡车在配送中心等待顾客下单,在  $h_0$  时刻出发,至少有 1 个订单出现,卡车离开配送中心。在后续调度中卡车在补货点完成补货后出发进行送货。之后调度中,卡车从补货点出发,依据  $R^*$  进行送货。

补货过程决策的内容涉及补货点  $m^*$  和调度时刻  $\mu^*$ 。卡车配送完所有车载订单之后接受补货无人机的补给,补货无人机在  $\mu^*$  时从配送中心出发前往  $m^*$  给卡车进行补货。由于不允许补货无人机先到达  $m^*$  等待卡车,因此  $\mu^*$  与当前配送路径的结束位置  $v$  和结束时间  $\theta_v$  关联。只有当补货无人机位于配送中心时,才能派遣补货无人机补货,因此决策在补货无人机返回配送中心的时刻  $h^U$  开始。

### 2.2 决策流程

除第 1 次调度外,每次调度需要决策  $m^*, \mu^*, R^*$ ,调度的决策取决于卡车的当前路线以及实时到达的订单  $\Omega_h$ 。根据  $h^U$  和当前配送路线上的最后一个交货点  $v$  的交货完成时刻  $\theta_v$  来确定每次调度决策开始的时间。在每次决策时,由于补货无人机前往每个补货点  $m \in M$  的最早出发时刻不相同,产生了许多可考虑的调度时刻,为便于决策,构建可考虑调度时刻的集合  $H = \{h_1, \dots, h_M\}$ 。构建过程如下:计算每个  $m \in M$  的最早的补货时刻  $\theta_m$ ,由  $\theta_m = \max\{\theta_v + t_{vm}^G, h^U + \bar{\tau} + t_{vOm}^{Us}\}$  确定。其中  $\theta_v + t_{vm}^G$  表示卡车完成  $v$  点的配送后立即前往  $m$  的到达时刻,  $h^U + \bar{\tau} + t_{vOm}^{Us}$  表示补货无人机从上次返回配送中心之后前往  $m$  的最早到达时刻,  $\bar{\tau}$  为补货无人机装货花费的时间。补货无人机前往  $m$  的最早调度时间,即从配送中心出发到  $m$  的最早时间为  $\mu_m = \theta_m - t_{vOm}^{Us}$ ,按照  $\mu_m, m \in M$  升序排列构建  $H = \{h_1, \dots, h_M\}$ 。补货无人机给卡车补货需要耗时  $\tau$ ,因此卡车在  $h + t_{vOm}^{Us} + \tau$  时刻出发开始新一轮配送。

决策具体流程如下:

步骤 1,输入  $h^U, v, \theta_v$ ;

步骤 2,构建  $H = \{h_1, \dots, h_M\}$ ;

步骤 3,选择  $H$  的首位  $h$  作为调度时间,更新  $\Omega_h$ ;

步骤 4,针对所有  $m \in M$ ,求解相应混合整数规划模型得到最优配送路径  $R_m$ ;

步骤 5,在所有  $R_m, m \in M$  中选择总成本最小的路径  $R_h$ ,识别补货点  $m_h$  和最早调度时间  $\mu_h$ ;

步骤 6,判断是否  $\mu_h \leq h$ ,是则执行步骤 7;否则  $H \leftarrow H \setminus \{h\}$ ,跳转至步骤 3;

步骤 7,  $R^* \leftarrow R_h, m^* \leftarrow m_h, \mu^* \leftarrow h$ 。

### 2.3 混合整数规划模型

决策的关键在于求解配送路径,配送路径影响了补货点的选择和未来的补货机会。本文建立混合整数规划模型,用于求解补货无人机出发时间为  $h$  时,在  $m$  补货后成本最小的配送路径  $R_m$ 。

#### 2.3.1 符号描述

表 1、表 2 对模型中出现的符号进行了描述性解释。

表 1 模型参数

Tab. 1 Model parameter table

参数	描述	参数	描述
$h$	补货无人机出发时间	$q_i$	$i$ 点的容量
$m$	补货点	$Q^{Uq}$	送货无人机的最大容量
$v_0$	配送中心	$D^{Uq}$	送货无人机最大行驶距离
$\Omega_h$	$h$ 时刻实时订单位置集, $i, j \in \Omega_h$	$\rho_i^o$	顾客 $i$ 的订单 $o$ 按时送达的最迟时间
$R$	车载无人机的数量	$M$	极大值
$K$	无人机的行程集合, $k \in K$	$C_1$	车辆的每公里行驶成本
$d_{ij}^U$	无人机从 $i$ 到 $j$ 行驶的路程	$C_2$	无人机每公里飞行成本
$d_{ij}^G$	卡车从 $i$ 到 $j$ 行驶的路程	$C_3$	延迟送达的惩罚成本
$t_{ij}^{Uq}$	送货无人机从 $i$ 到 $j$ 的行驶时间	$s^G$	卡车为客户提供服务的时间
$t_{ij}^G$	卡车从 $i$ 到 $j$ 的行驶时间	$s^{Uq}$	送货无人机为客户提供服务的时间

表 2 模型变量

Tab. 2 Model variable table

变量	变量类型	描述
$z_{ik}^U$	0-1 变量	无人机行程 $k$ 访问 $i$ 为 1, 否则为 0
$z_i^G$	0-1 变量	卡车访问 $i$ 为 1, 否则为 0
$x_{ij}$	0-1 变量	卡车从 $i$ 行驶到 $j$ 为 1, 否则为 0
$y_{ijk}$	0-1 变量	无人机行程 $k$ 从 $i$ 飞行到 $j$ 为 1, 否则为 0
$h_{ik}^R$	0-1 变量	无人机行程 $k$ 从 $i$ 降落为 1, 否则为 0
$h_{ik}^L$	0-1 变量	无人机行程 $k$ 从 $i$ 起飞为 1, 否则为 0
$f_i^G, f_i^U$	0-1 变量	卡车、无人机按时送达为 1, 否则为 0
$r_i$	非负整数变量	在 $i$ 点时卡车上空闲无人机的数量
$t_i^{Uq,A}, t_i^{Uq,L}$	连续变量	送货无人机访问 $i$ 的到达时间和离开时间
$t_i^{G,A}, t_i^{G,L}$	连续变量	卡车访问 $i$ 的到达时间和离开时间
$l_i^G, l_i^U$	连续变量	卡车配送点、无人机配送点的迟到时间

### 2.3.2 混合整数规划模型

模型以成本最小化为目标。在成本方面,不仅设置了卡车行驶成本和无人机行驶成本,还考虑了时效性影响,设置了延迟送达的惩罚成本。目标函数为:

$$\min C = C_1 \times \sum_{i \in \Omega_h \cup m} \sum_{j \in \Omega_h \cup v_0} d_{ij}^G \times x_{ij} + C_2 \times \sum_{k \in K} \sum_{i \in \Omega_h} \sum_{j \in \Omega_h} d_{ij}^U \times y_{ijk} + C_3 \times \sum_{i \in \Omega_h} \sum_{k \in K} z_{ik}^U \times l_i^U + C_3 \times \sum_{i \in \Omega_h} z_i^G \times l_i^G。$$

为确保卡车能正常行驶,保证每个客户都被服务,提出约束,见式(1)~(4)。约束(1)表示每个节点必须被访问且最多被访问 1 次;约束(2)表示车辆从补货点出发,最后回到配送中心;约束(3)强制卡车必须经过卡车配送的客户点;约束(4)保证卡车访问点的流平衡。

$$\sum_{k \in K} z_{ik}^U + z_i^G = 1, \forall i \in \Omega_h, \quad (1)$$

$$\sum_{j \in \Omega_h} x_{mj} = \sum_{i \in \Omega_h} x_{iv_0} = 1, \quad (2)$$

$$z_i^G = \sum_{j \in \Omega_h \cup m} x_{ji}, \forall i \in \Omega_h, \quad (3)$$

$$\sum_{j \in \Omega_h \cup v_0} x_{ij} = \sum_{j \in \Omega_h \cup m} x_{ji} \leq 1, \forall i \in \Omega_h。 \quad (4)$$

为防止无人机在不执行配送时沿着卡车路径进行不必要的飞行,提出约束确保无人机能正常地飞行,见式(5)~(10)。约束(5)保证每个无人机行程的节点包括起飞点和降落点的流平衡;约束(6)、(7)限制每个点的出入度,保证每个点只被配送 1 次;约束(8)表示每条路径最多被无人机访问 1 次;约束(9)保证由无人机配送的点必须有无人机访问;约束(10)确保无人机每次飞行至少有 1 个被配送的节点。

$$h_{ik}^R + \sum_{j \in \Omega_h} y_{ijk} = h_{ik}^L + \sum_{j \in \Omega_h \cup m} y_{jik}, \forall i \in \Omega_h, \forall k \in K, \quad (5)$$

$$\sum_{j \in \Omega_h} y_{ijk} \leq 1, \forall i \in \Omega_h \cup m, \forall k \in K, \quad (6)$$

$$\sum_{j \in \Omega_h \cup m} y_{jik} \leq 1, \forall i \in \Omega_h, \forall k \in K, \quad (7)$$

$$\sum_{k \in K} y_{ijk} + \sum_{k \in K} y_{jik} \leq 1, \forall i \in \Omega_h, \forall j \in \Omega_h, \quad (8)$$

$$\sum_{j \in \Omega_h \cup m} y_{jik} \geq z_{ik}^U, \forall i \in \Omega_h, \forall k \in K, \quad (9)$$

$$z_{ik}^U + z_{jk}^U \geq 1 - M(1 - y_{ijk}), \forall i \in \Omega_h, \forall j \in \Omega_h, \forall k \in K。 \quad (10)$$

为保证无人机能够正确地起飞和降落,提出约束,见式(11)~(16)。约束(11)、(12)分别确定每个行程  $k \in$

$K$  的起飞点和降落点;约束(13)、(14)确保所有起飞点和降落点都在卡车路径上;约束(15)约束同一行程的无人机起飞点和降落点不相同;约束(16)表示每个无人机行程的起飞点数量等于降落点,且最大为 1。

$$h_{ik}^L \geq 1 - M(2 - y_{ijk} - z_{jk}^U + z_{ik}^U), \forall i \in \Omega_h \cup m, \forall j \in \Omega_h, i \neq j, \forall k \in K, \quad (11)$$

$$h_{jk}^R \geq 1 - M(2 - y_{ijk} - z_{ik}^U + z_{jk}^U), \forall i \in \Omega_h, \forall j \in \Omega_h, i \neq j, \forall k \in K, \quad (12)$$

$$\sum_{k \in K} h_{ik}^L \leq M \sum_{j \in \Omega_h \cup m} x_{ji}, \forall i \in \Omega_h, \quad (13)$$

$$\sum_{k \in K} h_{ik}^R \leq M \sum_{j \in \Omega_h \cup v_0} x_{ij}, \forall i \in \Omega_h, \quad (14)$$

$$h_{ik}^L + h_{ik}^R \leq 1, \forall i \in \Omega_h, \forall k \in K, \quad (15)$$

$$\sum_{i \in \Omega_h \cup m} h_{ik}^L = \sum_{i \in \Omega_h} h_{ik}^R \leq 1, \forall k \in K. \quad (16)$$

考虑到配送无人机是同质的,有容量和续航的限制,并且在每次行程后更换电池可再次飞行,因此需要保证所有无人机行程都满足容量和续航约束,并且被可用无人机执行,为此提出了约束,见式(17)~(21)。约束(17)限制了无人机总数;约束(18)平衡了起飞、降落、空闲的无人机数量;约束(19)保证卡车在服务最后一个点时,将无人机全部收回;约束(20)限制每个无人机行程装载的货物不超过无人机容量;约束(21)限制每个无人机行程的飞行距离不超过无人机的续航限制。

$$r_m + \sum_{j \in \Omega_h} \sum_{k \in K} y_{mjk} = R, \quad (17)$$

$$r_i + \sum_{l \in \Omega_h \cup m} \sum_{k \in K} y_{ljk} \geq r_j + \sum_{l \in \Omega_h} \sum_{k \in K} y_{jlk} - M(1 - x_{ij}), \forall i \in \Omega_h \cup m, \forall j \in \Omega_h, i \neq j, \quad (18)$$

$$r_i \geq R - M(1 - x_{iv_0}), \forall i \in \Omega_h, \quad (19)$$

$$\sum_{i \in \Omega_h} z_{ik}^U \times q_i \leq Q^{Uq}, \forall k \in K, \quad (20)$$

$$\sum_{i \in \Omega_h \cup m} \sum_{j \in \Omega_h} d_{ij}^U \times y_{ijk} \leq D^{Uq}, \forall k \in K. \quad (21)$$

为消除卡车和无人机的子回路,并确定所有节点卡车或无人机的到达时间和离开时间,提出约束,见式(22)~(28)。约束(22)基于卡车从前一个节点的出发时间和相应的行驶时间来更新卡车在节点处的到达时间;约束(23)确保卡车在配送完成之前不能离开客户节点;约束(24)限制卡车在所有无人机在该节点处被取回之前离开该节点;约束(25)根据无人机从前一个节点的出发时间和相应的行驶时间来更新无人机在节点处的到达时间;约束(26)确保无人机在配送完成之前不能离开客户节点;约束(27),(28)用于保证卡车和无人机同时离开起飞点。

$$t_j^{G,A} \geq t_i^{G,L} + t_{ij}^G - M \times (1 - x_{ij}), \forall i \in \Omega_h \cup m, \forall j \in \Omega_h \cup v_0, \quad (22)$$

$$t_i^{G,L} \geq t_i^{G,A} + s^G - M \times (1 - z_i^G), \forall i \in \Omega_h, \quad (23)$$

$$t_i^{G,L} \geq t_i^{U,A} - M \times (1 - z_i^G), \forall i \in \Omega_h, \quad (24)$$

$$t_j^{Uq,A} \geq t_i^{Uq,L} + t_{ij}^{Uq} - M \times (1 - \sum_{k \in K} y_{ijk}), \forall i \in \Omega_h \cup m, \forall j \in \Omega_h, \quad (25)$$

$$t_i^{Uq,L} \geq t_i^{Uq,A} + s^{Uq} - M \times (1 - \sum_{k \in K} z_{ik}^U), \forall i \in \Omega_h, \quad (26)$$

$$t_i^{Uq,L} \geq t_i^{G,L} - M \times (1 - h_{ik}^L), \forall i \in \Omega_h \cup m, \forall k \in K, \quad (27)$$

$$t_i^{Uq,L} \leq t_i^{G,L} + M \times (1 - h_{ik}^L), \forall i \in \Omega_h \cup m, \forall k \in K. \quad (28)$$

由于订单在  $\rho_i^o$  之后送达被视为迟到,需要支付额外的惩罚成本,因此提出约束来确定每个客户点的迟到时间,见式(29)~(36)。式(29)~(32)计算了每个卡车配送节点的迟到时间;式(33)~(36)计算了每个无人机配送节点的迟到时间。

$$l_i^G \geq 0, \forall i \in \Omega_h, \quad (29)$$

$$l_i^G \geq t_i^{G,A} - \rho_i^o, \forall i \in \Omega_h, \quad (30)$$

$$l_i^G \leq M \times (1 - f_i^G), \forall i \in \Omega_h, \quad (31)$$

$$l_i^G \leq t_i^{G,A} - \rho_i^o + M \times f_i^G, \forall i \in \Omega_h, \quad (32)$$

$$l_i^U \geq 0, \forall i \in \Omega_h, \tag{33}$$

$$l_i^U \geq t_i^{U,A} - \rho_i^o, \forall i \in \Omega_h, \tag{34}$$

$$l_i^U \leq M \times (1 - f_i^U), \forall i \in \Omega_h, \tag{35}$$

$$l_i^U \leq t_i^{U,A} - \rho_i^o + M \times f_i^U, \forall i \in \Omega_h. \tag{36}$$

2.4 ALNSA

使用求解方案进行决策时需要多次求解路径,在应用中对决策的时效性要求很高。由于该问题为 NP-难,规模稍大时使用求解器求解混合整数规划模型耗时很长,为节约求解用时,本文针对该问题的特征,对算子进行了针对性的设计,提出了 ALNSA。使用 ALNSA 求出在  $m$  补货时  $\Omega_h$  内所有订单的配送成本最小的路径,算法总流程如图 2 所示。

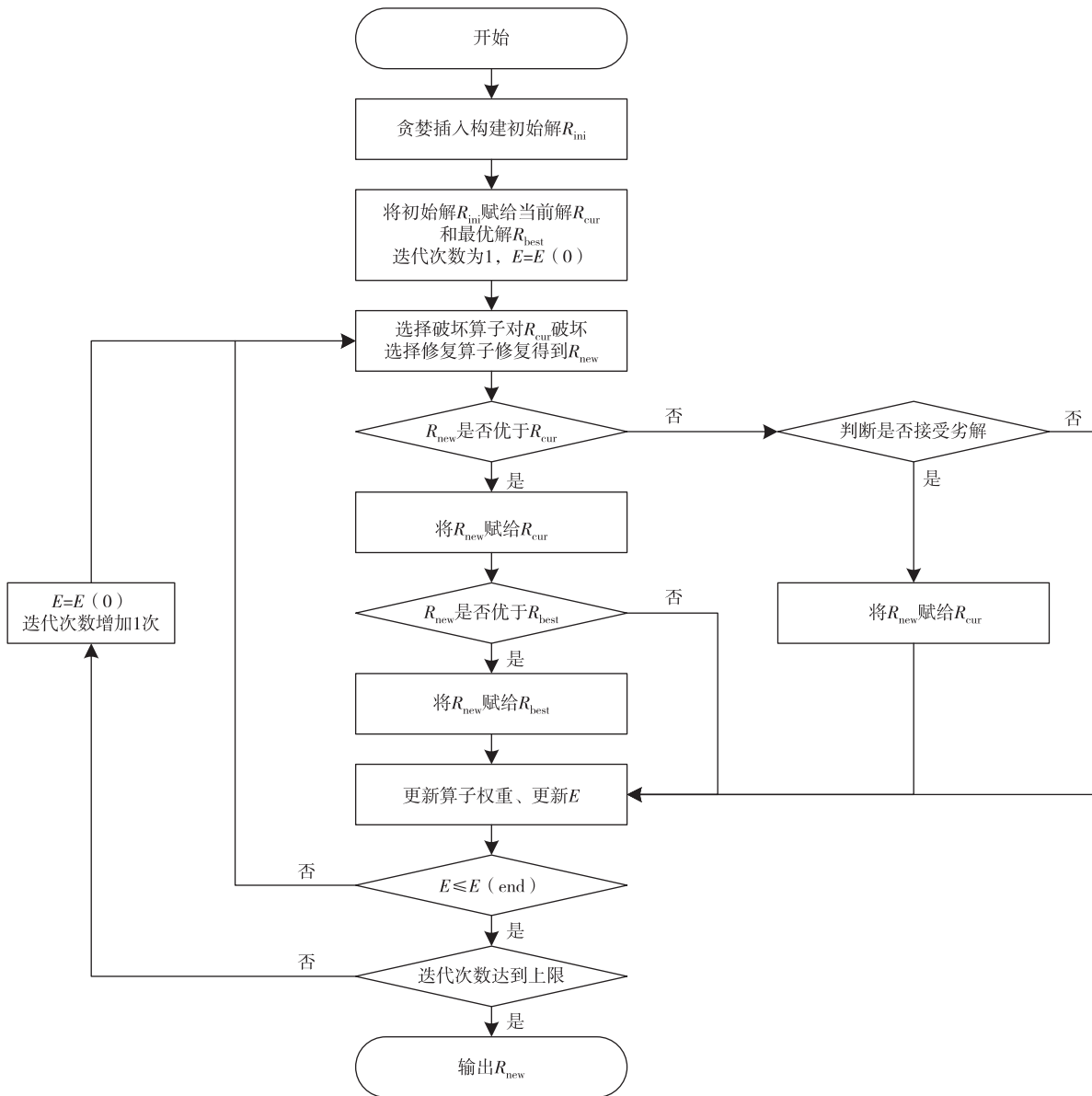


图 2 算法流程图

Fig. 2 Algorithm flow

2.4.1 构建初始解

初始解的优劣会影响 ALNSA 的求解效率,因此本文采用贪婪插入的思想构建初始解,具体步骤如下。

步骤 1,输入  $\Omega_h, m$  和送货开始时刻  $h + t_{v_0 m}^U + \tau$ 。

步骤 2,根据每个节点到  $m$  的距离将  $\Omega_h$  升序排列。

步骤 3,选择  $\Omega_h$  的首位客户点  $n$  直接加入路径  $R_{ini}$ ,  $\Omega_h \leftarrow \Omega_h \setminus \{n\}$ 。

步骤 4,  $\omega = \emptyset$ , 计算  $R_{ini}$  中每个卡车节点  $i$  的无人机可用数量  $r_i$ ,  $\omega \leftarrow \omega \cup \{r_i\}$ ,  $\omega$  按照  $R_{ini}$  中卡车访问顺序排列。

步骤 5,再次在  $\Omega_h$  选择  $n$ , 计算  $n$  插入  $R_{ini}$  中所有可插入位置  $l$  后的目标值  $O_l^n$ , 选择  $O_l^n$  中最小的  $B_l^n$ 。

步骤 6, 顺序判断是否  $r_i > 0$ , 顺序判断  $\omega$  中  $r_i$  后的节点  $z$  是否  $r_z > 0$ , 是则  $L \leftarrow L \cup \{z\}$ , 直到出现  $r_z = 0$ ,  $L \leftarrow L \cup \{z\}$ 。以  $i$  为起飞点,  $l$  中的节点  $j$  为降落点, 构建新飞行路径  $R_{ij}^f$ , 计算所有构建  $R_{ij}^f$  后的目标值  $O_{ij}^n$ 。  $R_{ij}^f$  超出  $C^{Uq}$  时,  $O_{ij}^n$  为无穷大。选择  $O_{ij}^n$  最小的  $B_{ij}^n$ 。

步骤 7, 比较  $B_l^n, B_{ij}^n$ 。如果  $B_l^n > B_{ij}^n$ , 将  $n$  插入  $R_{ini}$  中位置  $l$ , 否则, 在  $R_{ini}$  中加入  $R_{ij}^f$ 。  $\Omega_h \leftarrow \Omega_h \setminus \{n\}$ 。

步骤 8, 判断  $\Omega_h$  是否为空, 是则执行步骤 9, 否则执行步骤 4。

步骤 9, 输出  $R_{ini}$ 。

#### 2.4.2 算子设计

针对问题设计了 4 个破坏算子。

1) 贪婪无人机节点破坏算子: 贪婪破坏至多 3 个无人机配送节点。计算所有无人机配送节点移除之后改变的目标值, 移除对目标值影响最大的 1 个无人机配送节点, 将该节点加入被破坏节点集合中, 当该飞行路径中该点为唯一配送点时, 删除整条飞行路径。重复上述操作, 若无人机配送节点不足 3 个, 破坏所有无人机配送节点。

2) 贪婪卡车节点破坏算子: 贪婪破坏至多 3 个非起降点的卡车配送节点。计算所有非起降点卡车配送节点移除后改变的目标值, 移除对目标值影响最大的点并加入被破坏节点集合中。重复上述操作, 若操作时所有卡车节点皆为起降点, 则停止破坏。

3) 随机破坏非起飞降落点算子: 随机移除至多 3 个除起飞点降落点以外的任意节点。随机破坏有助于增加算法的随机性, 减低陷入局部最优的风险。

4) 随机破坏起飞降落点算子: 随机移除一段飞行路径的起飞点和降落点, 如果没有飞行路径则不进行破坏。由于起飞点和降落点一定是卡车路径中的点, 因此不把移除的点加入被破坏节点集合, 仅在飞行路径中删除起飞点和降落点。由于前 3 个破坏算子未涉及到起飞点和降落点的破坏, 因此创造了起飞点和降落点的优化空间, 同时增加了算法的随机性。

针对问题设计了 3 个修复算子。

1) 贪婪插入算子: 首先判断输入的被破坏路径中是否有破坏了起飞点和降落点的飞行路径。若有, 判断是否存在  $r_i > 0, i \in \Omega_h \cup m$ , 若存在, 则在所有  $i, r_i > 0$  中选择离飞行路径中第 1 个顾客点最近的点作为飞行路径的起飞点, 在该起飞点之后配送的卡车节点选择离飞行路径最后一个顾客点最近可行降落点作为飞行路径降落点, 否则将飞行路径中所有节点加入被破坏点集合。然后对被破坏点进行处理, 遍历所有可插入的位置, 计算插入后的目标值, 选择目标值最好的位置插入, 重复操作, 直到被破坏点全部修复完成。在遍历所有可插入位置计算目标值时, 如果存在  $r_i > 0, i \in \Omega_h \cup m$ , 给被破坏点创建新飞行路径也在被考虑的范围。

2) 随机插入算子: 首先判断有无被破坏起降点的飞行路径, 若存在可用的起降点, 则随机分配一组起降点给该飞行路径; 若不存在, 则将该飞行路径的所有节点加入被破坏点集合。将被破坏的节点随机插入可加入的位置, 同样包括创建新的飞行路径。如此做法增加算法的随机性, 避免陷入局部最优。

3) 后悔值插入算子: 首先判断输入的被破坏路径中是否有破坏了起飞点和降落点的飞行路径。若有, 判断是否存在  $r_i > 0, i \in \Omega_h \cup m$ , 若存在, 则根据  $r_i > 0, i \in \Omega_h \cup m$  创建所有可用的起飞点降落点组合的集合。遍历所有组合, 计算目标值  $O$ , 选择  $O$  最优的组合作为被破坏飞行路径的起飞点和降落点, 否则将飞行路径中所有节点加入被破坏点集合。然后对被破坏的顾客点处理, 计算所有被破坏点的后悔值  $\Delta O$ , 选择  $\Delta O$  最大的节点, 修复至最优位置。重新计算剩余点的  $\Delta O$ , 重复上述过程, 直到所有被破坏的节点被修复。后悔值是节点插入最优位置和次优位置的目标值的差值, 根据  $\Delta O = |O_1 - O_2|$  计算得出, 这里的位置同样包括了创建新的飞行路径。

#### 2.4.3 接受准则和自适应机制

为降低算法陷入局部最优解的风险, 引入基于模拟退火的评估标准来接受较差的解, 从而在全局上搜寻更

优的解。在 $[0, 1]$ 中随机生成 $\epsilon$ , 当 $\epsilon < P$ 时, 接受当前解, 当新解优于当前解时 $P = 1$ , 否则 $P = e^{-\frac{|O(\text{new}) - O(\text{cur})|}{E}}$ 。 $E$ 在每次迭代之后更新 $E(n+1) = \beta \times E(n)$ 。当 $E$ 减小到终止温度 $E(\text{end})$ 时, 退火完成, 开始新一轮迭代。

自适应机制的关键在于修改算子权重。算子的初始权重相同, 根据每次修复后得到的新解给算子打分。算子权重计算公式: $\psi_i = \alpha \times \psi_i + (1 - \alpha) A_i / B_i$ ,  $\psi_i$ 为算子 $i$ 的权重,  $\alpha$ 是权重更新系数,  $A_i$ 表示算子 $i$ 的总得分,  $B_i$ 表示算子 $i$ 的使用次数。算法根据新的权重去选择下一次破坏、修复算子。算子使用的结果越好, 得分和权重越高, 被选中的概率越大。

#### 2.4.4 自适应大邻域搜索流程

步骤 1, 当前解 $R_{\text{cur}} \leftarrow R_{\text{ini}}$ , 最优解 $R_{\text{best}} \leftarrow R_{\text{ini}}$ 。

步骤 2, 以算子权重 $\psi_i, \psi_j$ 作算子被选中概率, 选出破坏算子 $i$ 对 $R_{\text{cur}}$ 进行破坏, 选出修复算子 $j$ 对被破坏的解进行修复, 得到新解 $R_{\text{new}}$ 。

步骤 3, 计算 $R_{\text{new}}$ 的目标值 $O_{\text{new}}$ , 如果 $O_{\text{new}} < O_{\text{cur}}$ ,  $R_{\text{cur}} \leftarrow R_{\text{new}}$ , 执行步骤 4, 否则执行步骤 5。

步骤 4, 判断是否 $O_{\text{new}} < O_{\text{best}}$ , 是则 $R_{\text{best}} \leftarrow R_{\text{new}}$ , 算子得分为 $\sigma_1$ , 执行步骤 6; 否则算子得分为 $\sigma_2$ , 执行步骤 6。

步骤 5, 基于模拟退火的接受准则判断是否接受劣解, 随机生成 $\epsilon$ , 如果 $\epsilon < P$ , 则 $R_{\text{cur}} \leftarrow R_{\text{new}}$ , 算子得分为 $\sigma_3$ ; 否则不接受 $R_{\text{new}}$ , 算子得分为 $\sigma_4$ 。

步骤 6, 根据算子得分 $\sigma$ 更新算子总得分 $\Gamma \leftarrow \Gamma + \sigma$ , 更新算子权重 $\psi_i, \psi_j$ 。

步骤 7,  $E \leftarrow \beta \times E$ , 如果 $E < E(\text{end})$ , 则迭代次数增加 1 次, 执行步骤 7, 否则执行步骤 2。

步骤 8, 如果迭代次数到达上限,  $E \leftarrow E(0)$ , 执行步骤 2。否则结束搜索, 输出当前最优解。

### 3 算例实验

#### 3.1 实验设置

模型采用 Gurobi 10.0.1 求解, 所有求解方案均采用 Python 3.11 编写。所有计算过程均在配置为 Intel (R) Core(TM) i5-8300H CPU @ 2.30 GHz, 8 GB RAM, Windows 11 的计算机上运行。

根据 Solomon VRPTW 基准数据集 C101, C201, 按照相同序号间隔取 35 个节点位置, 在 35 个节点位置中, 按照相同间隔取 5 个位置作为补货点位置, 生成 $N_1 = 30, M_1 = 5$ 的节点规模 $V_1$ 和 $N_2 = 60, M_2 = 10$ 的节点规模 $V_2$ 。潜在客户在周期 $T$ 内下单( $T = 480 \text{ min}$ ), 下单的时间根据泊松过程随机生成, 订单货物质量在 $[0, 10]$ 之间随机生成。订单列表由客户点、下单时间、订单货物质量构成, 按照下单时间升序排列, 每个潜在客户的下单率为 $\lambda$ ,  $\lambda_1 = 0.015, \lambda_2 = 0.0075, \lambda_3 = 0.005$ 。算例命名规则为: 地图-节点规模-下单率, 例如 C1-1-1 即为 C101 中节点规模为 $V_1$ , 下单率为 $\lambda_1$ 的算例。配送中心位于坐标(50, 0)的位置, 节点之间的卡车路径采用曼哈顿距离, 无人机路径采用欧几里得距离。所有算例均采用求解方案进行决策, 求解路径部分除 3.2 中使用 Gurobi 求解器外, 均使用 ALNSA。其余参数取值见表 3。

#### 3.2 模型与算法的比较

在求解方案中分别使用 Gurobi 求解器和算法进行比较, 由于订单节点数较大时求解模型耗时较长, 选择较小的下单率 $\lambda_3 = 0.005$ 进行实验, 设置每次求解时间上限为 1 800 s, 时间上限内未解得最优解时输出当前最优解。设置本组实验的目的是检验算法的性能, 因此未把求解的耗时算入动态决策的时间内, 以免影响调度的时间, 从而影响订单的按时送达。CPU 时间为求解方案中调用模型或算法的 CPU 用时(单位: s); 送货成本包含了卡车和送货无人机从补货点开始送货到最后一个顾客点的所有行驶成本(单位: 元); 惩罚成本是指延迟送达产生的惩罚成本(单位: 元); 补货成本包含了卡车从最后一个顾客点到指定补货点的行驶成本和补货无人机从配送中心到指定补货点往返的行驶成本(单位: 元); 总成本为每次调度时的送货成本、惩罚成本及补货成本总和(单位: 元)。

求解结果如表 4 所示。在 C1-1-3 和 C2-1-3 算例中, 算法均能求解得出最优解, 模型和算法的每次调度决策均一致。在节点数较小时, 出现了 Gurobi 的 CPU 时间比 ALNSA 短的情况, 这是 ALNSA 即使在较小规模时依然要执行一定迭代次数所导致的, 在节点数稍大时, Gurobi 的 CPU 时间会远长于 ALNSA。在 C1-2-3 和 C2-2-3 算例中, 由于每次调度时订单节点数较大, 出现了多次到达求解时间上限的情况, 算法与之相比不仅节约了大量的求解时间, 还在 C1-2-3 的第 6 次调度以及 C2-2-3 的第 5 次调度出现了比 Gurobi 求解 1 800 s 更优的解。

表 3 参数设置表  
Tab. 3 Parameter setting table

参数	取值	描述
$v^G$	4	卡车速度,模拟现实场景 $48 \text{ km} \cdot \text{h}^{-1}$
$v^{Uq}$	6	送货无人机速度,模拟现实场景 $72 \text{ km} \cdot \text{h}^{-1}$
$v^{Us}$	5	补货无人机速度,模拟现实场景 $60 \text{ km} \cdot \text{h}^{-1}$
$C^{Uq}$	75	送货无人机最远续航里程,模拟 $15 \text{ km}$ 续航
$Q^{Uq}$	15	送货无人机最大容量,单位:件
$Q^{Us}$	200	补货无人机最大容量,单位:件
$C_1$	0.3	卡车单位里程行驶成本,模拟 $1.5 \text{ 元} \cdot \text{km}^{-1}$
$C_2$	0.02	送货无人机单位里程行驶成本,模拟 $0.1 \text{ 元} \cdot \text{km}^{-1}$
$C_3$	1.5	每分钟迟到时间的惩罚成本, $1.5 \text{ 元} \cdot \text{min}^{-1}$
$C_4$	0.03	补货无人机单位里程行驶成本,模拟 $0.15 \text{ 元} \cdot \text{km}^{-1}$
$R$	2	车载无人机架数
$S$	60	订单服务保障时间,单位: min
$s^G$	1	卡车服务客户耗时,单位: min
$s^{Uq}$	1	送货无人机服务客户耗时,单位: min
$h_0$	30	卡车从配送中心出发的时刻
$\bar{\tau}$	2	补货无人机装货耗时,单位: min
$\underline{\tau}$	2	卡车装货耗时,单位: min
$E(0)$	100	模拟退火初始温度,单位: $^{\circ}\text{C}$
$E(\text{end})$	10	模拟退火终止温度,单位: $^{\circ}\text{C}$
$\beta$	0.97	退火速率
$\sigma_1, \dots, \sigma_4$	1.5, 1.2, 0.8, 0.6	算子的得分
$\alpha$	0.5	权重更新系数
$\Lambda$	100	最大迭代次数

### 3.3 无人机补货效果分析

通过比较无人机同时用于补货、送货与无人机仅用于送货这 2 种情况,分析无人机补货的效果。无人机仅用于配送时,卡车每次配送完装载订单之后直接返回配送中心补货,再重新出发,求解结果如表 5 所示:无人机参与补货平均节约成本 37.80%。支出较少的补货成本就能使得总成本减少较多,配送中心在(50,0)时,平均每支出 1 单位补货成本能节约总成本 20.40 元,在算例 C1-1-2 中尤为明显,总成本减少了 48.44%。

为进一步分析无人机补货的效果,考虑了配送中心距离城市中心较远的情况,将配送中心位置设置为(50,-25)进行算例实验。与配送中心在(50,0)的情况相比,配送中心距离城市中心较远时无人机参与补货的效果更加显著,平均节约成本 45.75%,平均每支出 1 单位补货成本能节约总成本 31.65 元。

### 3.4 无人机送货效果分析

为分析无人机配送的效果,将卡车上搭载的送货无人机数量  $R$  分别设置为 0, 2 和 4 进行算例实验。 $R=0$  时送货过程仅由卡车单独完成,补货过程依旧由补货无人机完成。如表 6 所示:送货过程使用无人机能够有效提高配送效率,大幅度减少了迟到现象。在  $R=0$  和  $R=2$  的对比中,2 架配送无人机的加入使得总成本平均减少了 80.63%,惩罚成本平均减少了 85.42%,大幅度提高了送货的及时性;在  $R=0$  和  $R=4$  的对比中,无人机送货的效果具有明显优势,总成本的平均减幅达到了 86.22%,惩罚成本平均减少了 91.53%;相较于  $R=2$ ,  $R=4$  的总成本平均降低了 30.85%。 $R$  从 2 增加到 4 带来的收益小于  $R$  从 0 增加到 2,存在边际效用递减现象。这是因为在订单数不太多的情况下,不需要同时使用 4 架无人机也能高效地完成配送任务。考虑到给卡车配备无人机的固定成本并未计入算例的成本中,因此配置的无人机并非越多越好,需要根据实际订单量情况决定。

表 4 Gurobi 求解与 ALNSA 求解结果对比

Tab. 4 Comparison of results of Gurobi solution and ALNSA

算例	调度	订单量	Gurobi					ALNSA					GAP
			CPU 时间/s	送货 成本/元	惩罚 成本/元	补货 成本/元	总成本/ 元	CPU 时间/s	送货 成本/元	惩罚 成本/元	补货 成本/元	总成本/ 元	
C1-1-3	1	3	0.06	32.78	0	0	32.78	4.45	32.78	0	0	32.78	0
	2	3	0.16	38.02	0	10.75	48.77	6.34	38.02	0	10.75	48.77	0
	3	11	6.69	47.55	6.23	12.92	66.70	5.56	47.55	6.23	12.92	66.70	0
	4	9	15.37	50.62	21.26	11.65	83.53	6.94	50.62	21.26	11.65	83.53	0
	5	9	2.02	36.06	0	7.73	43.79	6.98	36.06	0	7.73	43.79	0
	6	7	5.20	47.63	16.9	10.75	75.28	9.04	47.63	16.90	10.75	75.28	0
	7	5	0.24	46.94	0.40	6.50	53.84	1.30	46.94	0.40	6.50	53.84	0
	8	2	0.02	5.88	0	7.73	13.61	1.24	5.88	0	7.73	13.61	0
	9	2	0.02	2.48	0	7.73	10.21	1.77	2.48	0	7.73	10.21	0
	10	2	0.02	42.59	0	5.33	47.92	1.80	42.59	0	5.33	47.92	0
	11	3	0.02	41.15	0	7.45	48.60	4.42	41.15	0	7.45	48.60	0
总计	56		391.70	44.79	88.54	525.03		391.70	44.79	88.54	525.03	0	
C2-1-3	1	2	0.02	30.29	0	0	30.29	1.90	30.29	0	0	30.29	0
	2	9	3.25	39.34	0	11.48	50.82	11.30	39.34	0	11.48	50.82	0
	3	10	45.19	77.23	66.93	4.06	148.22	21.80	77.23	66.93	4.06	148.22	0
	4	14	65.45	56.11	189.58	5.25	250.94	20.18	56.11	189.58	5.25	250.94	0
	5	9	22.35	21.39	7.20	9.68	38.27	10.37	21.39	7.20	9.68	38.27	0
	6	7	2.22	51.72	4.21	7.36	63.29	8.67	51.72	4.21	7.36	63.29	0
	7	5	0.26	24.52	0	2.40	26.92	3.59	24.52	0	2.40	26.92	0
	8	4	0.04	38.49	0	12.08	50.57	2.73	38.49	0	12.08	50.57	0
	9	6	0.32	33.13	0	2.40	35.53	5.97	33.13	0	2.40	35.53	0
	10	1	0.01	49.19	0	5.69	54.88	1.51	49.19	0	5.69	54.88	0
总计	67		421.41	267.92	60.4	749.73		421.41	267.92	60.4	749.73	0	
C1-2-3	1	7	1.53	37.81	0	0	37.81	9.82	37.81	0	0	37.81	0
	2	8	3.30	39.58	0	3.90	43.48	12.37	39.58	0	3.90	43.48	0
	3	13	1 800	37.79	10.14	4.74	52.67	16.80	38.15	10.27	4.74	53.16	0
	4	13	1 800	61.47	27.47	4.74	93.68	18.08	61.47	27.47	4.74	93.68	0
	5	14	1 800	71.12	99.52	3.59	174.23	23.03	71.12	99.52	3.59	174.23	0
	6	28	1 800	71.82	852.12	8.10	932.04	41.45	57.87	515.89	8.10	581.86	-37.57%
	7	18	1 800	41.54	78.51	13.66	133.71	25.15	41.54	78.51	13.66	133.71	0
	8	20	1 800	59.01	85.12	6.04	150.17	37.90	59.01	85.12	6.04	150.17	0
	9	8	30.18	43.29	63.06	8.61	114.96	15.51	43.29	63.06	8.61	114.96	0
总计	129		463.43	1 215.94	53.38	1 732.75		449.84	879.84	53.38	1 383.06	-20.18%	
C2-2-3	1	13	45.52	59.03	7.19	0	66.22	20.66	59.03	7.19	0	66.22	0
	2	17	1 800	44.30	16.16	9.18	69.64	22.32	44.30	16.16	9.18	69.64	0
	3	17	1 800	60.21	14.68	3.30	78.19	26.93	60.21	14.68	3.30	78.19	0
	4	22	1 800	82.89	163.98	8.56	255.43	68.28	82.89	163.98	8.56	255.43	0
	5	19	1 800	64.24	101.68	8.56	174.48	64.63	52.93	100.02	8.56	161.51	-7.43%
	6	13	1 800	36.25	0	3.30	39.55	33.50	36.25	0	3.30	39.55	0
	7	15	226.05	40.30	0	7.25	47.55	48.17	40.30	0	7.25	47.55	0
	8	6	3.49	56.18	0	5.56	61.74	10.19	56.18	0	5.56	61.74	0
总计	122		443.40	303.69	45.71	792.80		432.09	302.03	45.71	779.83	-1.64%	

注: GAP 为 ALNSA 与 Gurobi 的总成本之差除以 Gurobi 的总成本, 再乘 100%; 求解时间内未得出最优解时, 取当前最优。

表 5 无人机补货效果分析表  
Tab. 5 UAVs replenishment effect analysis

配送中心位置	算例	订单总量/单	无人机仅送货			无人机同时补货和送货					GAP	
			调度次数	送货成本/元	惩罚成本/元	总成本/元	调度总数/元	送货成本/元	惩罚成本/元	补货成本/元		总成本/元
(50, 0)	C1-1-1	195	7	591.09	3 104.29	3 695.38	9	441.34	2 253.98	78.80	2 774.12	-24.93%
	C2-1-1	190	6	603.54	4 017.56	4 621.10	9	513.82	2 273.90	59.94	2 847.66	-38.38%
	C1-1-2	101	7	540.35	1 270.12	1 810.47	12	408.28	430.29	94.83	933.40	-48.44%
	C2-1-2	92	6	557.40	1 060.82	1 618.22	9	489.67	576.48	64.82	1 130.97	-30.11%
	C1-2-2	189	6	560.07	4 326.15	4 886.22	8	459.71	2 283.73	48.10	2 791.54	-42.87%
	C2-2-2	194	6	532.95	4 265.65	4 798.60	9	476.34	2 249.34	53.99	2 779.67	-42.07%
(50, -25)	C1-1-1	195	6	596.10	3 883.17	4 479.27	9	477.62	2 416.07	75.86	2 969.55	-33.70%
	C2-1-1	190	5	581.56	5 450.83	6 032.39	9	517.09	2 861.77	71.32	3 450.18	-48.81%
	C1-1-2	101	6	552.85	2 355.52	2 908.37	10	474.83	708.90	69.43	1 253.16	-56.91%
	C2-1-2	92	5	532.34	2 193.03	2 725.37	9	516.83	902.80	71.24	1 490.87	-45.30%
	C1-2-2	189	5	578.20	6 645.75	7 223.95	9	485.82	3 362.52	64.81	3 913.15	-45.83%
	C2-2-2	194	6	636.77	5 977.57	6 614.34	9	495.27	2 738.04	76.87	3 310.18	-49.95%

注:GAP 为无人机补货与无人机仅配送的总成本之差除以无人机仅配送成本再乘 100%。

表 6 无人机送货效果分析表  
Tab. 6 UAVs delivery effect analysis

R	算例	订单总量/单	调度次数	送货成本/元	惩罚成本/元	补货成本/元	总成本/元	GAP(0)	GAP(2)
0	C1-1-1	195	5	576.90	11 700.81	27.71	12 305.42		
	C2-1-1	190	5	568.20	11 642.89	32.13	12 243.22		
	C1-1-2	101	6	564.90	6 334.35	32.18	6 931.43		
	C2-1-2	92	6	518.40	4 378.71	35.52	4 932.63		
	C1-2-2	189	5	564.90	18 019.59	29.51	18 614.00		
	C2-2-2	194	5	549.00	14 009.55	30.89	14 589.44		
2	C1-1-1	195	9	441.34	2 253.98	78.80	2 774.12	-77.46%	
	C2-1-1	190	9	513.82	2 273.90	59.94	2 847.66	-76.74%	
	C1-1-2	101	12	408.28	430.29	94.83	933.40	-86.53%	
	C2-1-2	92	9	489.67	576.48	64.82	1 130.97	-77.07%	
	C1-2-2	189	8	459.71	2 283.73	48.10	2 791.54	-85.00%	
	C2-2-2	194	9	476.34	2 249.34	53.99	2 779.67	-80.95%	
4	C1-1-1	195	9	512.81	1 541.09	73.01	2 126.91	-82.72%	-23.33%
	C2-1-1	190	8	504.69	1 616.26	78.52	2 199.47	-82.04%	-22.76%
	C1-1-2	101	13	419.89	267.19	97.76	784.84	-88.68%	-15.92%
	C2-1-2	92	9	511.33	538.86	63.38	1 113.57	-77.42%	-1.54%
	C1-2-2	189	10	468.62	444.57	54.80	967.99	-94.80%	-65.32%
	C2-2-2	194	11	478.03	665.64	72.84	1 216.51	-91.66%	-56.24%

注:GAP(0)为 R 与 R 为 0 的总成本之差除以 R 为 0 的总成本再乘 100%,GAP(2)同理。

### 3.5 服务保障时间的影响

上述算例实验中出现了较多的延迟交付情况,导致惩罚成本较高。为探究服务保障时间产生的影响,将服务保障时间 S 分别设置为 60,90 和 120 进行对比,求解结果见表 7。与 S=60 相比,S=90 大幅度减少了成本,C1-2-2 最为突出,减少了 79.93%。服务保障时间宽松后,更多订单能够按时送达,大幅度减少了惩罚成本。在

S=90 和 S=120 的比较中,订单数较少的 C1-1-2、C2-1-2 成本节约相较于其他算例不明显,这是由于大部分订单能够在 90 min 内送达,不需要增加更多的服务保障时间。通过对成本分析还发现,给予更宽松的时间限制,不仅能减少大部分惩罚成本,还能做出行驶距离更短的决策,在 S=60 和 S=120 的比较中行驶成本平均节约了 12.48%。

表 7 服务保障时间分析表  
Tab. 7 Service support time analysis

S	算例	订单总量/单	调度次数	送货成本/元	惩罚成本/元	补货成本/元	总成本/元	GAP(60)	GAP(90)
60	C1-1-1	195	9	441.34	2 253.98	78.8	2774.12		
	C2-1-1	190	9	513.82	2 273.9	59.94	2 847.66		
	C1-1-2	101	12	408.28	430.29	94.83	933.4		
	C2-1-2	92	9	489.67	576.48	64.82	1 130.97		
	C1-2-2	189	8	459.71	2 283.73	48.10	2 791.54		
	C2-2-2	194	9	476.34	2 249.34	53.99	2 779.67		
90	C1-1-1	195	8	452.79	150.71	54.91	658.41	-76.27%	
	C2-1-1	190	8	525.42	343.42	51.35	920.19	-67.69%	
	C1-1-2	101	11	417.16	26.39	83.57	527.12	-43.53%	
	C2-1-2	92	8	512.60	44.19	40.62	597.41	-47.18%	
	C1-2-2	189	9	427.99	79.24	53.09	560.32	-79.93%	
	C2-2-2	194	8	423.26	111.36	56.84	591.46	-78.72%	
120	C1-1-1	195	8	377.39	12.59	70.03	460.01	-83.42%	-30.13%
	C2-1-1	190	7	438.08	4.46	57.20	499.74	-82.45%	-45.69%
	C1-1-2	101	9	416.62	0	66.10	482.72	-48.28%	-8.42%
	C2-1-2	92	8	460.96	0	50.45	511.41	-54.78%	-14.40%
	C1-2-2	189	7	375.38	3.97	40.25	419.60	-84.97%	-25.11%
	C2-2-2	194	8	383.59	0	55.14	438.73	-84.22%	-25.82%

注:GAP(60)为 S 与 S 为 60 的总成本之差除以 S 为 60 的总成本再乘 100%,GAP(90)同理。

### 3.6 决策目标的影响

通过之前算例实验分析发现,一次调度的配送用时对下次补货配送的及时性产生较大影响,因此设置不同的目标函数进行对比。如表 8 所示,将目标改成最小化配送时间对算例进行求解,结果表明以最小化配送时间为目标的决策在整体上不仅不能减少延迟交付,还加剧了迟到现象。这是因为在以配送用时为目标的决策中,快迟到的订单并未被优先考虑,以设置了惩罚成本的成本最小化为目标能够兼顾配送及时性和行驶成本,使得快迟到的订单能够被优先送达,在整体上实现了更好的配送效果。

表 8 决策目标分析表  
Tab. 8 Decision objective analysis

算例	订单		成本				配送时间					GAP
	总量/单	调度次数	送货成本/元	惩罚成本/元	补货成本/元	总成本/元	调度次数	送货成本/元	惩罚成本/元	补货成本/元	总成本/元	
C1-1-1	195	9	441.34	2 253.98	78.8	2 774.12	8	463.61	2 962.3	57.82	3 483.73	25.58%
C2-1-1	190	9	513.82	2 273.9	59.94	2 847.66	8	545.33	3 007.88	50.56	3 603.77	26.55%
C1-1-2	101	12	408.28	430.29	94.83	933.40	12	455.77	530.28	86.75	1 072.80	14.93%
C2-1-2	92	9	489.67	576.48	64.82	1 130.97	9	536.02	1 045.87	57.28	1 639.17	44.93%
C1-2-2	189	8	459.71	2 283.73	48.10	2 791.54	8	499.56	2 897.10	49.58	3 446.24	23.45%
C2-2-2	194	9	476.34	2 249.34	53.99	2 779.67	9	515.33	2 907.8	55.85	3 478.98	25.16%

注:GAP 等于以“时间”为目标函数的总成本与以“成本”为目标函数的点成本之差除以以“成本”为目标函数的总成本再乘 100%。

## 4 总结

本文提出了当日达模式下考虑无人机补货的车机协同配送问题,将无人机既用于给卡车实时补货又用于与卡车协同送货,使当日达配送成本低且高效。设计了基于枚举和混合整数规划模型的求解方案,用于决策每次调度的补货点、调度时刻和配送路径。其中混合整数规划模型以最小化成本为目标,考虑了无人机的最大续航里程、最大容量,允许货物延迟送达并据此设置了惩罚成本。为提高求解效率,结合问题特点设计了 ALNSA,对算子进行了针对性改进,并通过算例实验验证了算法的有效性。

通过一系列算例实验得出以下结论:

1) 无人机既用于给卡车实时补货又用于与卡车协同送货的方式能够有效降低当日达的配送成本、提高当日达的配送及时性。相较于无人机仅用于配送的情况平均节约总成本 37.80%,减少惩罚成本 46.17%;相较于无人机仅补货平均节约成本 80.63%,减少惩罚成本 85.42%。

2) 无人机补货省去了卡车返回配送中心补货再出发的过程,因此当配送中心远离城市中心时,采用无人机补货的效果更好。

3) 卡车配置的无人机数量对配送效果的影响存在边际效益递减规律,现实应用中可以根据订单数量来配置卡车上的无人机。

4) 服务保障时间变长不仅能大幅度减少惩罚成本也能使行驶成本减少,也存在边际效益递减规律。

5) 相较于以最小化配送时间为目标,以惩罚成本、行驶成本之和最小化为目标的求解方案能优先考虑快迟到订单,提高配送的及时性,做出整体配送效果更好的决策。

在现实应用中,存在客户要求的配送时效不同的情况,可以考虑提出服务保障时间不同的选项供客户选择。当一次送货过程的耗时太长,会产生货物无法及时补给到卡车上的情况,导致货物无法按时送达,在后续研究中,可以考虑完工时间最小化和成本最小化的多目标优化。

### 参考文献:

- [1] 中华人民共和国国家邮政局. 国家邮政局公布 2023 年邮政行业运行情况[EB/OL]. (2024-01-22)[2024-04-22]. <https://www.spb.gov.cn/gjyzj/c100015/c100016/202401/59eeb6e8b0e7404f8127aa2c7aebded6.shtml>.  
State Post Bureau of the People's Republic of China. The State Post Bureau releases the operational status of the postal industry for 2023 [EB/OL]. (2024-01-22)[2024-04-22]. <https://www.spb.gov.cn/gjyzj/c100015/c100016/202401/59eeb6e8b0e7404f8127aa2c7aebded6.shtml>.
- [2] ZHANG J, VAN WOENSEL T. Dynamic vehicle routing with random requests: a literature review[J]. *International Journal of Production Economics*, 2023, 256: 108751.
- [3] AZI N, GENDREAU M, POTVIN J Y. A dynamic vehicle routing problem with multiple delivery routes[J]. *Annals of Operations Research*, 2012, 199(1): 103-112.
- [4] KLAPP M A, ERERA A L, TORIELLO A. The one-dimensional dynamic dispatch waves problem[J]. *Transportation Science*, 2016, 52(2): 402-415.
- [5] KLAPP M A, ERERA A L, TORIELLO A. The dynamic dispatch waves problem for same-day delivery[J]. *European Journal of Operational Research*, 2018, 271(2): 519-534.
- [6] KLAPP M A, ERERA A L, TORIELLO A. Request acceptance in same-day delivery[J]. *Transportation Research Part E: Logistics and Transportation Review*, 2020, 143: 102083.
- [7] ULMER M W, STRENG S. Same-day delivery with pickup stations and autonomous vehicles[J]. *Computers & Operations Research*, 2019, 108: 1-19.
- [8] DAYARIAN I, SAVELSBERGH M, CLARKE J P. Same-day delivery with drone resupply[J]. *Transportation Science*, 2020, 54(1): 229-249.
- [9] CHEN X W, ULMER M W, THOMAS B W. Deep Q-learning for same-day delivery with vehicles and drones[J]. *European Journal of Operational Research*, 2022, 298(3): 939-952.
- [10] MURRAY C C, CHU A G. The flying sidekick traveling salesman problem: optimization of drone-assisted parcel delivery[J]. *Transportation Research Part C: Emerging Technologies*, 2015, 54: 86-109.
- [11] AGATZ N, BOUMAN P, SCHMIDT M. Optimization approaches for the traveling salesman problem with drone[J].

- Transportation Science, 2018, 52(4):965-981.
- [12] POIKONEN S, GOLDEN B, WASIL E A. A branch-and-bound approach to the traveling salesman problem with a drone[J]. *INFORMS Journal on Computing*, 2019, 31(2):335-346.
- [13] ES YUREK E, OZMUTLU H C. A decomposition-based iterative optimization algorithm for traveling salesman problem with drone[J]. *Transportation Research Part C: Emerging Technologies*, 2018, 91:249-262.
- [14] HA Q M, DEVILLE Y, PHAM Q D, et al. A hybrid genetic algorithm for the traveling salesman problem with drone[J]. *Journal of Heuristics*, 2020, 26(2):219-247.
- [15] PENG K, DU J X, LU F, et al. A hybrid genetic algorithm on routing and scheduling for vehicle-assisted multi-drone parcel delivery[J]. *IEEE Access*, 2019, 7:49191-49200.
- [16] MOSHREF-JAVADI M, HEMMATI A, WINKENBACH M. A truck and drones model for last-mile delivery: a mathematical model and heuristic approach[J]. *Applied Mathematical Modelling*, 2020, 80:290-318.
- [17] BOYSEN N, BRISKORN D, FEDTKE S, et al. Drone delivery from trucks: drone scheduling for given truck routes [J]. *Networks*, 2018, 72(4):506-527.
- [18] HU M L, LIU W D, LU J Q, et al. On the joint design of routing and scheduling for Vehicle-Assisted Multi-UAV inspection [J]. *Future Generation Computer Systems*, 2019, 94:214-223.
- [19] MURRAY C C, RAJ R. The multiple flying sidekicks traveling salesman problem: parcel delivery with multiple drones[J]. *Transportation Research Part C: Emerging Technologies*, 2020, 110:368-398.
- [20] POIKONEN S, GOLDEN B. Multi-visit drone routing problem[J]. *Computers & Operations Research*, 2020, 113:104802.
- [21] KITJACHAROENCHAI P, MIN B C, LEE S. Two echelon vehicle routing problem with drones in last mile delivery[J]. *International Journal of Production Economics*, 2020, 225:107598.
- [22] LUO Z H, POON M, ZHANG Z Z, et al. The multi-visit traveling salesman problem with multi-drones[J]. *Transportation Research Part C: Emerging Technologies*, 2021, 128:103172.
- [23] PINA-PARDO J C, SILVA D F, SMITH A E. The traveling salesman problem with release dates and drone resupply[J]. *Computers & Operations Research*, 2021, 129:105170.
- [24] PINA-PARDO J C, SILVA D F, SMITH A E, et al. Fleet resupply by drones for last-mile delivery [J]. *European Journal of Operational Research*, 2024, 316(1):168-182.
- [25] PINA-PARDO J C, SILVA D F, SMITH A E, et al. Dynamic vehicle routing problem with drone resupply for same-day delivery [J]. *Transportation Research Part C: Emerging Technologies*, 2024, 162:104611.

## Optimization Algorithms and Its Applications

### Truck-Drone Collaborative Distribution with Drone Resupply in the Same-Day Delivery

MA Yunfeng<sup>1,2</sup>, OUYANG Lijun<sup>1</sup>, HU Jian<sup>1</sup>, HU Yina<sup>1</sup>, REN Liang<sup>1,2</sup>

(1. School of Management, Wuhan University of Science and Technology;

2. Center for Service Science and Engineering, Wuhan University of Science and Technology, Wuhan 430065, China)

**Abstract:** E-commerce companies have implemented the same-day delivery model to meet customers' demand for prompt courier services. To enhance delivery efficiency and reduce costs, the use of drones in collaboration with truck delivery and replenishment is studied. Divide a complete delivery cycle into multiple scheduling decisions based on the completion time of the delivery, and the solution of each scheduling decision is based on enumeration. The problem is decomposed into multiple mixed integer programming subproblems, and the efficiency of solving the subproblems is improved by using an improved adaptive large neighborhood search algorithm. Experimental results demonstrate that the enhanced adaptive large neighborhood search algorithm significantly reduces solving time without compromising solution quality. Furthermore, compared with using UAVs solely for replenishment or delivery purposes, the simultaneous utilization of UAVs for both replenishment and delivery effectively enhances distribution efficiency while reducing overall costs.

**Keywords:** same-day delivery; drone resupply; truck-drone collaborative distribution

(责任编辑 黄 颖)