

doi:10.19920/j.cnki.jmsc.2025.01.005

# 考虑无人机站的车辆路径规划问题<sup>①</sup>

方 侃<sup>1,2</sup>, 安义丹<sup>1</sup>, 朱 宁<sup>1</sup>, 黄 典<sup>1,2,3\*</sup>

(1. 天津大学管理与经济学部, 天津 300072; 2. 天津大学复杂管理系统实验室, 天津 300072;  
3. 天津大学新媒体与传播学院, 天津 300072)

**摘要:**近年来,无人机调度受到了学者们的日益关注.本研究提出了考虑无人机站的车辆路径规划问题,以总成本最低为目标函数,考虑了包裹的最晚送达时间、无人机的数量限制和最长可持续飞行时间等约束条件.针对该问题本研究首先构建了一类基于弧的原问题模型.随后,使用 Dantzig-Wolfe 分解,将原问题模型分解为基于路径的限制主问题模型和卡车子问题模型以及无人机子问题模型,并设计了分支定价算法进行求解,以获得全局最优解.大量的数值实验结果表明分支定价算法在求解时间和求解质量上均优于主流商业求解器.此外,本研究发现卡车的行驶成本在总成本中占据较高比例.同时,相对于无人机的数量和无人机与卡车的速度比而言,无人机可访问的顾客的比例对总成本有显著影响.因此,扩大无人机的适用范围,可以有效地降低总成本.

**关键词:**无人机站;无人机调度;车辆路径规划问题;分支定价

**中图分类号:** C931      **文献标识码:** A      **文章编号:** 1007-9807(2025)01-0061-16

## 0 引 言

近年来,电子商务行业蓬勃发展<sup>[1,2]</sup>,超过 80% 以上的包裹可由无人机配送<sup>[3]</sup>,物流配送被认为是无人机最有发展前景的应用领域之一<sup>[4]</sup>.目前,用于物流配送的无人机多采用电池推进系统,具有速度快、造价低和使用方便等优点.相对而言,传统的卡车配送方式往往伴随更多的成本和污染<sup>[5,6,7]</sup>.因此,许多企业近年来开始推出一类可自动化完成包裹收寄与存储、无人机暂存与升降和无人机电池更换与充电等功能的无人机站<sup>[8]</sup>.无人机站不仅可以使无人机的使用更加高效便利,也进一步降低了人工运营成本.混合智能管理系统是未来的发展重点<sup>[9]</sup>,然而目前考虑无人机站的调度优化研究相对较少.因此,如何设计

一种低成本有效率的调度方案,使得无人机与卡车在无人机站的协助下能够更好地完成配送任务,具有重要的理论和实践意义.

目前,关于无人机的路径优化问题受到了研究者的广泛关注. Murray 和 Chu<sup>[10]</sup>将无人机加入到旅行商问题 (traveling salesman problem, TSP) 中,首次提出带有飞行伙伴的旅行商问题 (flying side-kick TSP, FSTSP) 和考虑并行无人机调度的旅行商问题 (parallel drone scheduling TSP, PDSTSP). 在 FSTSP 中,考虑了一辆卡车和一架无人机,要求卡车与无人机高度同步.其中,无人机由卡车携带,并可以在某些地点启动自动送货,完成要求的订单后,返回卡车.在 PDSTSP 中,则考虑了一辆卡车和多架无人机,无人机的使用效率高度依赖于仓库与顾客的相对位置.其中,无人机可

① 收稿日期: 2021-12-26; 修订日期: 2023-02-15.

基金项目: 国家自然科学基金资助项目 (72471169; 72401214; 72231005; 72122015; 71971154); 教育部人文社会科学研究项目 (24YJC630078); 天津市哲学社会科学规划项目 (TJGL21-016); 天津市自然科学基金资助项目 (23JCQNJC01900); 天津大学复杂管理系统实验室开放课题.

通讯作者: 黄 典 (1989—), 男, 安徽六安人, 博士, 讲师. Email: huangdian@tju.edu.cn

被用于服务仓库附近的顾客,从仓库出发访问顾客之后再返回仓库,而卡车则单独行驶访问顾客,二者同时进行.与 FSTSP 相比,Agatz<sup>[11]</sup>等提出的带有无人机的旅行商问题 (traveling salesman problem with drone, TSP-D) 要求无人机的速度比卡车更快,且卡车可在原地等待无人机完成任务. Murray 和 Chu<sup>[10]</sup>提出的 FSTSP 与 PDSTSP 和 Agatz<sup>[11]</sup>等提出的 TSP-D,成为后续若干卡车与无人机协作调度优化问题研究的基础.

近年来,许多学者对 TSP-D 的约束条件进行了扩展,提出了不同的变体问题并设计了相应的算法. Carlsson 和 Song<sup>[12]</sup>假设无人机可在卡车行驶的任何地方离开卡车. Murray 和 Raj<sup>[13]</sup>假设一辆卡车可携带多架无人机. Boysen 等<sup>[14]</sup>考虑了卡车按照固定路线行驶场景下的无人机调度. Jeong 等<sup>[15]</sup>考虑了包裹重量和禁飞区域对无人机调度的影响.从算法角度而言,目前关于此类问题精确算法方面的研究仍较为缺乏. Bouman 等<sup>[16]</sup>针对 TSP-D 提出了精确的动态规划算法. Roberti 和 Ruthmair<sup>[17]</sup>则为 TSP-D 的变体,设计了分支定价算法.另一方面,不少研究者针对这类问题提出了若干改进的启发式算法.例如 Freitas 和 Penna<sup>[18]</sup>提出了混合启发式算法,以 TSP 的解决方案作为初始解,再使用变邻域搜索来获得最终路线. Ha 等<sup>[19]</sup>对 Murray 和 Chu<sup>[10]</sup>提出的启发式算法进行修正,并提出了贪婪随机自适应算法. Poikonen 等<sup>[20]</sup>假设卡车可以在无人机交付时停止等待,并提出四种基于分支定界算法框架的启发式算法. Tu 等<sup>[21]</sup>进一步为此类问题提出了一类自适应大邻域搜索算法.

随着研究的不断深入,关于无人机的调度优化开始从一辆车与无人机协作的方式扩展到多辆车与无人机协作的方式,由 TSP-D 扩展为带有无人机的车辆路径规划问题 (vehicle routing problem with drone, VRP-D). Wang 等<sup>[22]</sup>提出了 VRP-D,考虑用配备无人机的卡车为顾客配送包裹,其中,假设无人机可在仓库或任何顾客点升降,卡车和无人机都使用欧氏距离,且电池拥有无限的电量.针对这一问题,其对几种最坏的情况进行分析.在前文的基础上, Wang 等<sup>[23]</sup>进一步考虑卡车遵循曼哈顿距离,无人机遵循欧式距离,且电池容量有限等条件,并针对不同目标函数的 VRP-D 进

行了理论分析.

基于 Wang 等<sup>[23]</sup>的工作,学者们通过添加新的约束或考虑新的变体,来扩展 VRP-D 模型. Schermer 等<sup>[24]</sup>允许无人机在同一顶点起飞和降落. Wang 和 Sheu<sup>[25]</sup>引入无人机一次飞行中可访问多个顾客,设置服务中心作为无人机起飞和降落的站点,并证明了延长无人机的最大持续飞行时间可以使得总成本降低. Schermer 等<sup>[26]</sup>假设无人机可以在路径中的某些非顾客点起飞和降落,提出了带有途中操作的 VRP-D. Karak 和 Abdelghany<sup>[27]</sup>将 VRP-D 扩展为混合卡车-无人机路径问题,假设每架无人机每次飞行可进行多个取货和送货操作. Sacramento 等<sup>[28]</sup>考虑了卡车和无人机完成任务后可同时或分开返回仓库的场景,以降低运营成本为目标,设计了自适应大邻域搜索算法. Tamke 和 Buscher<sup>[29]</sup>考虑将卡车和无人机作为串联系统,并设计分支切割法进行求解.

如前文所述,无人机站作为卡车与无人机的中介,可以降低卡车与无人机之间的同步性要求,减少卡车在等待无人机上所花费的时间.与传统仓库相比,专用的无人机站对水电、网络和交通的要求更低,不仅有效地降低了成本<sup>[30]</sup>,而且可以满足交通不便地区客户的包裹运输需求.目前,考虑无人机站的研究仍较为缺乏.在 Murray 和 Chu 所提及的 PDSTSP 中事实上也考虑了将仓库视为单独的无人机站的情况.针对该问题,研究者亦做了一些相关的工作. Mbiadou 等<sup>[31]</sup>将 PDSTSP 划分为两阶段问题,提出了一种迭代的两阶段启发式算法,第一阶段使用动态规划有效地完成卡车和无人机之间的顾客划分,第二阶段优化卡车和无人机的路径. Dell'Amico 等<sup>[32]</sup>针对 PDSTSP 将整数规划与局部搜索相结合,并与 Murray 和 Chu<sup>[10]</sup>和 Mbiadou 等<sup>[31]</sup>所提的方法进行对比,证明了其在计算效率和有效性方面均表现更好.另一方面, Kim 和 Moon<sup>[33]</sup>引入了位于顾客中心的无人机站,允许货物暂存和无人机的升降,且假设足够多数量的无人机.类似地, Li 等<sup>[34]</sup>考虑了包含一个主要配送中心和多个仓库的三层配送系统,对成本进行分析,证实了使用卡车和无人机联合的送货系统比只使用卡车更为有效. Schermer 等<sup>[35]</sup>以总运行时间最短为目标,综合考虑无人机

站选择、卡车路线规划以及无人机调度的优化问题,证明了使用无人机站在减少交付时间上的有效性。

综上所述,学者们针对卡车和无人机结合的车辆路径规划问题进行了较多的研究,其中多数以卡车作为无人机的升降平台,从而扩大无人机的飞行范围,并利用两者之间的协作完成最后一公里配送。显然,这些优化模型对卡车和无人机之间的同步性要求较高。另一方面,针对使用无人机站的研究则相对较少,且目前已有的研究往往假设无人机站中无人机的数量足够多且无人机仅可飞行一次,较少考虑实际配送情景中无人机数量有限且无人机可多次飞行的场景<sup>[10, 33, 34]</sup>。此外,大多数学者往往仅对卡车与无人机结合的问题设计了一些启发式算法,对可求得最优解的精确式算法的研究相对较少<sup>[10, 18, 28]</sup>。基于上述情况,本

研究提出了考虑无人机站的车辆路径规划问题(vehicle routing problem with drone stations, VRP-DS),通过引入无人机站,在一个集成优化模型中综合考虑车辆调度、无人机分配和无人机调度等决策问题,从而实现在给定的时间范围内完成包裹配送,并使总成本最小化。针对该类问题,本研究首先为其构建了一类基于弧的原问题模型。基于该问题的特征,本研究进一步使用 Dantzig-Wolfe 分解将原问题模型分解为基于路径的限制主问题模型、卡车子问题模型和无人机子问题模型,并为其设计了一类分支定价算法以求得全局最优解。最后,通过大量算例实验,验证了文中所构造的模型和分支定价算法的有效性,并对关键因素进行敏感度分析,据此提出相关管理启示,从而为相关物流配送企业使用无人机站提供决策参考。本研究的技术框架如图1所示。

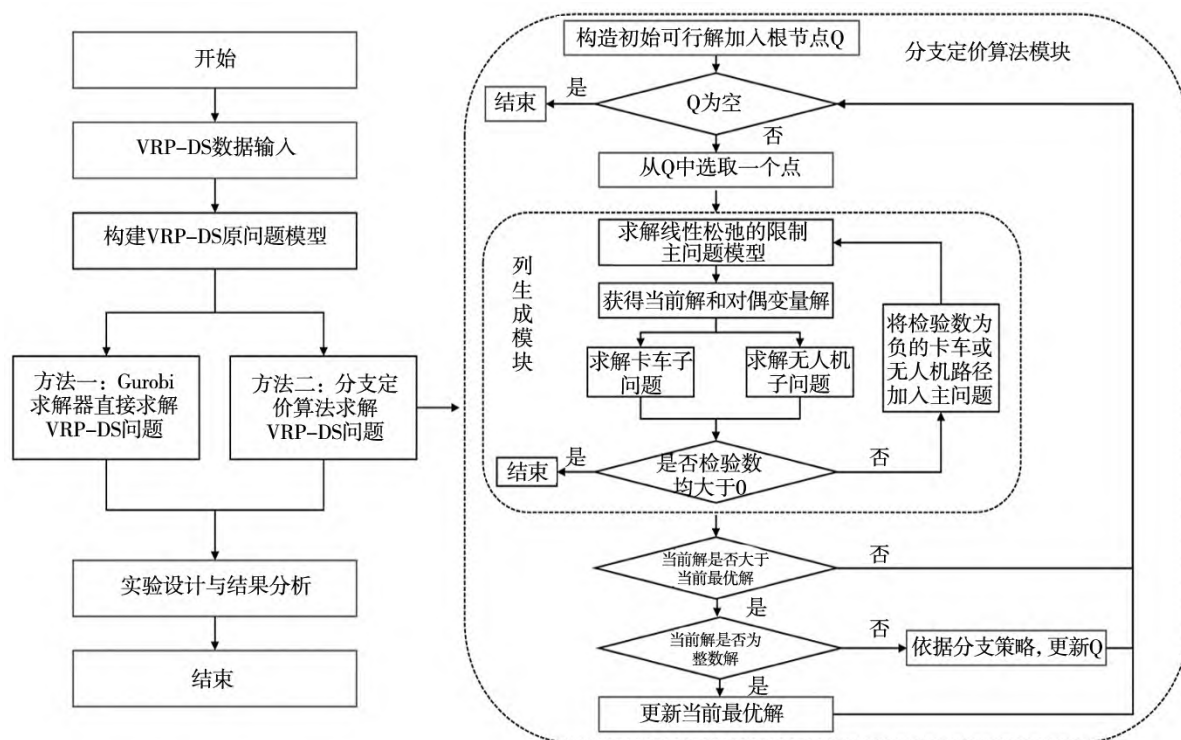


图1 技术框架示意图

Fig. 1 Schematic diagram of technical framework

## 1 问题与模型

### 1.1 问题描述

本研究考虑如下的问题:在VRP-DS中,包含一个仓库和一组无人机站,在仓库中停放了若干

辆同质的卡车和若干架同质的无人机。VRP-DS要求提供总成本最低的车辆和无人机调度方案,来满足无人机站的需求和顾客的需求。为此,卡车从仓库出发,可直接访问顾客或无人机站,最终返回仓库。卡车在访问无人机站时,可卸载部分包裹和无人机,由无人机站派遣无人机满足顾客的需求。



关于 VRP-DS 问题,本研究做出以下几点假设:

- 1) 顾客的需求不能分割,且需在最晚送达时间之前被满足;
- 2) 卡车容量足够大,与无人机互不干扰;
- 3) 无人机站容量足够大. 一个无人机站可同时发射多架无人机,且可被多辆卡车同时访问. 无人机站的需求,必须由卡车来满足;
- 4) 无人机每次完成配送任务后,需返回其所服务的无人机站;
- 5) 无人机每次往返耗时不能超过其最大飞行时间.

为了阐明上述问题描述,图 2 给出了一个简单的 VRP-DS 实例,其中包含一个仓库、三个无人机站,三架无人机和四辆卡车. 在该图中,中间的大方框表示仓库;无人机站编号为 1-3;顾客编号为 4-16,其中方形为只能由卡车访问的顾客,圆形为卡车与无人机均可访问的顾客;实线表示卡车行驶路径;虚线表示无人机飞行路径. 以卡车 3 为例说明其运行情况,卡车 3 依次访问顾客 10,无人机站 2,并在无人机站 2 中留下 2 架无人机和部分包裹,再由无人机 6 访问顾客 11,无人机 5 访问顾客 12 和 13,同时卡车 3 访问顾客 14 后返回仓库.

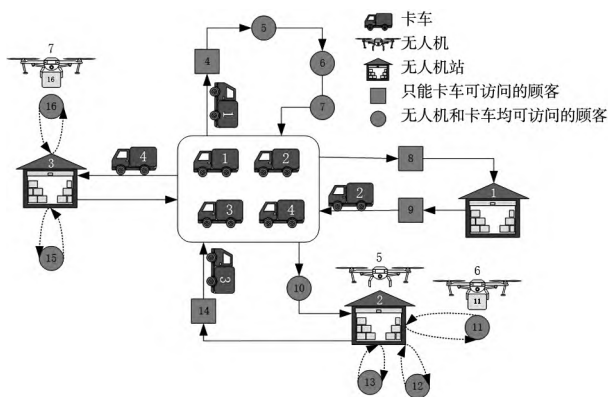


图 2 VRP-DS 的一个实例

Fig. 2 An example of VRP-DS

## 1.2 参数符号

本研究将 VRP-DS 定义在一个有向图  $G = (N, A)$  上,其中  $N$  是所有点的集合,  $A$  是所有弧的集合. 为了后续建模方便,本研究为仓库设置了两个不同的符号, 0 和  $n+1$ , 分别表示起始站和终止站,其中  $n$  是所有无人机站和顾客的总数量. 同样地,本研究将每个无人机站  $k \in K$  进行复制,并

将其复制点记作  $k'$ ,  $k$  作为无人机站  $k$  处的无人机首次飞行的起点,  $k'$  作为无人机站  $k$  处的无人机末次返回的终点,卡车无需访问复制的无人机站. 本研究将所有顾客  $C$  分为两类,只能被卡车访问的顾客记为  $C_1$ ,无人机和卡车均可访问的顾客记为  $C_2$ . 本研究所使用的符号定义具体如表 1 所示.

表 1 符号定义

Table 1 Symbol definitions

符号	定义
$C$	所有的顾客集合
$C_1$	只能被卡车访问的顾客集合
$C_2$	无人机和卡车均可访问的顾客集合
$K$	无人机站的集合
$K'$	有需求的无人机站的集合
$N_k^0$	无人机站点 $k \in K$ 处的无人机所有的可能出发点, $N_k^0 = C_2 \cup \{k\}$
$N_k^+$	无人机站点 $k \in K$ 处的无人机所有的可能到达点, $N_k^+ = C_2 \cup \{k'\}$
$N$	所有点的集合, $N = C \cup K \cup \{0, n+1\}$
$N^0$	卡车所有的可能出发点, $N^0 = N \setminus \{n+1\}$
$N^+$	卡车所有的可能到达点, $N^+ = N \setminus \{0\}$
$A$	所有弧的集合
$T$	卡车的集合
$D'$	卡车的数量
$D$	无人机的集合
$D^d$	无人机的数量
$l_i$	顾客 $i \in C$ 的最晚访问时间
$\theta_i$	每个点 $i \in N$ 的服务时间
$Q$	无人机单次最大可持续飞行时间
$d_{ij}^t$	卡车经过弧 $(i, j) \in A$ 的距离
$d_{ij}^d$	无人机经过弧 $(i, j) \in A$ 的距离
$\tau_{ij}^t$	卡车经过弧 $(i, j) \in A$ 的旅行时间
$\tau_{ij}^d$	无人机经过弧 $(i, j) \in A$ 的旅行时间
$F^t$	卡车的固定成本
$C^t$	卡车的每单位路程的运输成本
$C^d$	无人机的每单位路程的运输成本
$R^t$	可行的卡车路径的集合
$R_k^d$	无人机站点 $k \in K$ 处可行的无人机路径集合
$c_p^t$	卡车行驶路径 $p \in R^t$ 的行驶成本
$c_{ks}^d$	无人机从 $k \in K$ 出发完成路径 $s \in R_k^d$ 的行驶成本
$a_{pj}$	卡车路径 $p \in R^t$ 是否访问节点 $j$ , 若访问则为 1, 否则为 0
$b_{ksj}$	无人机路径 $s \in R_k^d$ 是否访问节点 $j$ , 若访问则为 1, 否则为 0
$t_{rj}$	路径 $r \in R^t \cup R_k^d$ 第一次访问节点 $j \in N$ 的时间, 当 $r \in R^t$ 时, 以卡车从仓库出发为零时刻, 当 $r \in R_k^d$ 时, 以无人从无人站 $k$ 出发为零时刻

### 1.3 原问题模型

本研究定义以下决策变量:  $u_v^t \in \{0,1\}$ , 若使用卡车  $v \in T$  则为 1, 否则为 0;  $u_{kv'}^d \in \{0,1\}$ , 若无人机站  $k \in K$  使用无人机  $v' \in D$  则为 1, 否则为 0;  $x_{ijv} \in \{0,1\}$ , 若卡车  $v \in T$  经过弧  $(i,j) \in A$  则为 1, 否则为 0;  $y_{kijv'} \in \{0,1\}$ , 若无人机  $k \in K$  使用无人机  $v' \in D$  先访问点  $i \in N_k^0$ , 返回无人机站  $k$  后, 接着访问点  $j \in N_k^+$  则为 1, 否则为 0;  $t_i$  表示点  $i \in N$  第一次被卡车或无人机访问的时间. 基于上述定义, 构建 VRP-DS 的原问题模型

$$\min \sum_{v \in T} F^t \cdot u_v^t + \sum_{v \in T} \sum_{i \in N^0} \sum_{j \in N^+} C^t \cdot d_{ij}^t x_{ijv} + \sum_{v' \in D} \sum_{k \in K} \sum_{i \in N_k^0} \sum_{j \in C_2} 2 \cdot C^d \cdot d_{kj}^d \cdot y_{kijv'} \quad (1)$$

$$\text{s. t. } \sum_{j \in N^+} x_{0jv} \leq u_v^t \quad \forall v \in T \quad (2)$$

$$\sum_{j \in N^+} x_{0jv} = \sum_{j \in N^0} x_{j,n+1,v} \quad \forall v \in T \quad (3)$$

$$\sum_{i \in N^0} x_{ijv} = \sum_{h \in N^+} x_{jihv} \quad \forall j \in C \cup K, v \in T \quad (4)$$

$$\sum_{i \in N^0} \sum_{v \in T} x_{ijv} = 1 \quad \forall j \in C_1 \quad (5)$$

$$\sum_{i \in N^0} \sum_{v \in T} x_{ikv} \geq 1 \quad \forall k \in K' \quad (6)$$

$$t_j - t_i - \theta_i + M \left( 1 - \sum_{v \in T} x_{ijv} \right) \geq \tau_{ij}^t \quad \forall i \in N^0, j \in N^+ \quad (7)$$

$$\sum_{i \in N^0} \sum_{v \in T} x_{ikv} \geq u_{kv'}^d \quad \forall k \in K, v' \in D \quad (8)$$

$$\sum_{k \in K} u_{kv'}^d \leq 1 \quad \forall v' \in D \quad (9)$$

$$\sum_{j \in N_k^+} y_{kijv'} \leq u_{kv'}^d \quad \forall k \in K, v' \in D \quad (10)$$

$$\sum_{j \in N_k^+} y_{kijv'} = \sum_{i \in N_k^0} y_{kik'v'} \quad \forall k \in K, v' \in D \quad (11)$$

$$\sum_{i \in N_k^0} y_{kijv'} = \sum_{h \in N_k^+} y_{kjhv'} \quad \forall k \in K, j \in C_2, v' \in D \quad (12)$$

$$2\tau_{kj}^d + \theta_j + M \left( 1 - \sum_{v' \in D} y_{kijv'} \right) \leq Q \quad \forall k \in K, i \in N_k^0, j \in N_k^+ \quad (13)$$

$$\sum_{i \in N^0} \sum_{v \in V} x_{ijv} + \sum_{k \in K} \sum_{i \in N_k^0} \sum_{v' \in V} y_{kijv'} = 1 \quad \forall j \in C_2 \quad (14)$$

$$t_j - t_i - \theta_i + M \left( 1 - \sum_{v' \in D} y_{kijv'} \right) \geq \tau_{ki}^d + \tau_{kj}^d$$

$$\forall k \in K, i \in N_k^0, j \in N_k^+ \quad (15)$$

$$t_i \leq l_i \quad \forall i \in C \quad (16)$$

$$0 \leq t_i \quad \forall i \in N \quad (17)$$

$$x_{ijv} \in \{0,1\} \quad \forall i \in N^0, j \in N^+, v \in T \quad (18)$$

$$y_{kijv'} \in \{0,1\} \quad \forall v' \in D, k \in K, i \in N_k^0, j \in N_k^+ \quad (19)$$

式(1)是最小化总成本, 总成本包含卡车固定成本、卡车运输成本和无人机运输成本.

式(2)~式(7)是卡车路径约束. 式(2)约束了卡车只有被使用才可以运送包裹, 式(3)保证了仓库是卡车的起点和终点. 式(4)确保卡车路径的流平衡. 式(5)保证了只能由卡车访问的顾客点被卡车访问且仅访问一次. 式(6)确保有需求的无人机站必须由卡车访问至少一次. 式(7)用于计算卡车访问相应点的时间, 且保证卡车路线中无子圈.

式(8)~式(13)是无人机路径约束. 式(8)确保无人机站被卡车访问后, 才可派遣无人机出发访问顾客. 式(9)保证了一架无人机只能分配给一个无人机站. 式(10)保证了无人机只有被使用才可以运送货物, 式(11)保证无人机站是无人机的起终点. 式(12)确保无人机路线的流平衡. 式(13)确保了无人机一次飞行不超过其最大可持续飞行时间.

式(14)~式(15)用于建立卡车和无人机之间的协作关系. 式(14)表示卡车和无人机均可访问的顾客被卡车或无人机访问且仅访问一次. 式(15)用于计算无人机访问相应点的时间, 并保证无人机路线中无子圈. 式(16)为最晚访问时间约束. 式(17)~式(19)为变量类型约束.

### 1.4 基于路径的限制主问题模型

由于 Lenstra 和 Kan<sup>[36]</sup> 已经证明了 VRP 是 NP-hard 问题, 且 VRP-DS 显然包含了 VRP, 所以 VRP-DS 也属于 NP-hard 问题. 此外, VRP-DS 的原问题模型中包含大量的变量和复杂约束, 因此, 可以预见随着问题规模的增大, 直接求解该优化模型可能效率较低. 特别需要注意的是, 在原问题模型中, 并非所有约束都包含全部变量, 而是存在一部分约束仅包含卡车相关变量, 呈现“对角线分块”结构. 基于上述观察, 本研究采用 Dantzig-Wolfe 分解, 将 VRP-DS 的原问题模型, 分解为基

于路径的限制主问题模型、卡车子问题模型和无人机子问题模型.

具体而言,限制主问题模型包含一组可行的卡车路径和无人机路径,其目标是寻找一组满足卡车与无人机协作约束的最优路径子集,使得总成本最低.为此,本研究定义以下决策变量:  $x_p \in \{0,1\}$ ,若选择卡车路径  $p \in R^t$  则为 1,否则为 0;  $y_{ks} \in \{0,1\}$ ,若选择无人机站点  $k \in K$  处的无人机路径  $s \in R_k^d$  则为 1,否则为 0;  $z_k$  表示无人机站点  $k$  分配的无人机数量,即无人机站点  $k \in K$  可选择的无人机路线的总数量.基于上述定义,构建了基于路径的限制主问题模型

$$\min \sum_{p \in R^t} F^t \cdot x_p + \sum_{p \in R^t} c_p^t \cdot x_p + \sum_{k \in K} \sum_{s \in R_k^d} c_{ks}^d \cdot y_{ks} \quad (20)$$

$$\text{s. t. } D^t - \sum_{p \in R^t} x_p \geq 0 \quad (21)$$

$$\sum_{p \in R^t} a_{pj} \cdot x_p = 1 \quad \forall j \in C_1 \quad (22)$$

$$\sum_{p \in R^t} a_{pk} \cdot x_p \geq 1 \quad \forall k \in K' \quad (23)$$

$$D^d \cdot \sum_{p \in R^t} a_{pk} \cdot x_p - z_k \geq 0 \quad \forall k \in K \quad (24)$$

$$D^d - \sum_{k \in K} z_k \geq 0 \quad (25)$$

$$z_k - \sum_{s \in R_k^d} y_{ks} \geq 0 \quad \forall k \in K \quad (26)$$

$$\sum_{p \in R^t} a_{pj} \cdot x_p + \sum_{k \in K} \sum_{s \in R_k^d} b_{ksj} \cdot y_{ks} = 1 \quad \forall j \in C_2 \quad (27)$$

$$l_j - \left( \sum_{p \in R^t} t_{pk} \cdot x_p + \sum_{s \in R_k^d} t_{sj} \cdot y_{ks} \right) +$$

$$M \left( 1 - \sum_{s \in R_k^d} b_{ksj} \cdot y_{ks} \right) \geq 0$$

$$\forall k \in K, j \in C_2 \quad (28)$$

$$x_p \in \{0,1\} \quad \forall p \in R^t \quad (29)$$

$$y_{ks} \in \{0,1\} \quad \forall k \in K, s \in R_k^d \quad (30)$$

$$z_k \in N \quad \forall k \in K \quad (31)$$

式(20)为最小化总成本,与式(1)等价.式(21)约束卡车路径总量不超过卡车数量.式(22)表示只能由卡车访问的顾客点,必须由卡车访问一次,与式(5)等价.式(23)表示卡车必须访问有需求的无人机站,与式(6)等价.式(24)表示无人机站被卡车访问后,才可调度无人机.式(25)表示选择的无人机路径总量不超过无人机数量.式(26)表示每个无人机站所选择的无人机路径总量不超

过分配给该无人机站的无人机数量.式(27)表示可被卡车或无人机访问的顾客点需被无人机或卡车访问且仅访问一次,与式(14)等价.式(28)表示被无人机访问的顾客的最晚送达时间约束,与式(16)等价.式(29)~式(31)为变量类型约束.

由于上述限制主问题模型中的决策变量包含整数变量,这使得其求解比较复杂.因此,本研究对式(29)~式(31)进行线性松弛来简化主问题的求解,线性松弛后的约束条件如式(32)~式(34)所示.因此,线性松弛的限制主问题模型的目标函数为式(20),约束条件为式(21)~式(27)和式(32)~式(34).

$$0 \leq x_p \leq 1 \quad \forall p \in R^t \quad (32)$$

$$0 \leq y_{ks} \leq 1 \quad \forall s \in R_k^d, k \in K \quad (33)$$

$$0 \leq z_k \quad \forall k \in K \quad (34)$$

在线性松弛的限制主问题模型中,无需列举所有路径,仅考虑可能改善当前最优解的路径即可.VRP-DS是最小化问题,所以寻找改善当前最优解的路径可转变为寻找检验数为负的路径,即卡车子问题和无人机子问题的目标函数为最小化检验数.为了描述检验数,本研究使用  $\pi^1$ 、 $\pi^2$ 、 $\pi^3$ 、 $\pi^4$ 、 $\pi^5$ 、 $\pi^6$ 、 $\pi^7$ 、 $\pi^8$  来表示对偶变量,分别对应于式(20)~式(27).

### 1.5 卡车子问题模型

卡车子问题的目标是生成检验数为负的满足卡车约束的卡车路径.本研究定义  $RC_i$  为卡车路径变量  $x_p$  的检验数.

$$RC_i = F^t + c_p^t + \pi^1 - \sum_{j \in C_1} \pi_j^2 a_{jp} - \sum_{k \in K'} \pi_k^3 a_{kp} - \sum_{k \in K} \pi_k^4 D^d a_{kp} - \sum_{j \in C_2} \pi_j^7 a_{jp} + \sum_{k \in K} \sum_{j \in C_2} \pi_{kj}^8 t_{kp} \quad (35)$$

本研究定义以下决策变量:  $x_{ij} \in \{0,1\}$ ,若卡车经过弧  $(i,j) \in A$  则为 1,否则为 0;  $t_i$  表示卡车到达节点  $i \in N$  的时间.以最小化  $RC_i$  为目标函数,考虑原问题模型中与卡车路径相关的约束式(2)~式(6)构建卡车子问题模型  $S_v^{truck}$ ,如式(36)~式(43)所示

$$\min F^t + \sum_{i \in N^0} \sum_{j \in N^+} c_{ij}^t x_{ij} + \pi^1 - \sum_{j \in C_1} \sum_{i \in N^0} \pi_j^2 x_{ij} - \sum_{k \in K'} \sum_{i \in N^0} \pi_k^3 x_{ik} - \sum_{k \in K} \sum_{i \in N^0} \pi_k^4 D^d x_{ik} -$$

$$\sum_{j \in C_2} \sum_{i \in N^0} \pi_j^7 x_{ij} + \sum_{k \in K} \sum_{j \in C_2} \pi_{jk}^8 t_k \quad (36)$$

$$\text{s. t. } \sum_{j \in N^+} x_{0j} = 1 \quad (37)$$

$$\sum_{j \in N^0} x_{j,n+1} = 1 \quad (38)$$

$$\sum_{i \in N^0} x_{ij} = \sum_{h \in N^+} x_{jh} \quad \forall j \in C \quad (39)$$

$$t_j - t_i - \theta_i + M(1 - x_{ij}) \geq \tau_{ij}^l \quad \forall i \in N^0, j \in N^+ \quad (40)$$

$$t_i \leq l_i \quad \forall i \in C \quad (41)$$

$$0 \leq t_i \quad \forall i \in N \quad (42)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in N^0, j \in N^+ \quad (43)$$

式(37)和式(39)保证了卡车的流量平衡,与式(2)~式(4)等价. 式(40)计算了卡车访问相应点的时间,避免了子圈的产生,与式(19)对应. 式(41)为最晚送达时间约束,与式(16)对应. 式(42)和式(43)为变量类型约束.

### 1.6 无人机子问题模型

不同的无人机站,对应不同的无人机子问题. 无人机子问题的目标是生成检验数为负的满足无人机路径约束的无人机路径. 本研究定义  $RC_{kd}$  为无人机路径变量  $y_{ks}$  的检验数,如式(44)所示.

$$RC_{kd} = c_{ks}^d + \pi_k^6 - \sum_{j \in C_2} \pi_j^7 b_{ksj} + \sum_{j \in C_2} \pi_{jk}^8 (t_{js} + Mb_{ksj}) \quad (44)$$

本研究定义以下决策变量:  $y_{ij} \in \{0, 1\}$ , 若无人机先访问顾客  $i$  后,返回无人机站  $k$  后,接着访问顾客  $j$ , 则为 1, 否则为 0.  $t_i$  表示无人机到达点  $i$  的时间,以无人机第一次从无人机站起飞为零时刻. 基于上述定义,构建无人机子问题模型  $S_{kv'}^{drone}$ , 如式(46)和式(52)所示.

$$\min \sum_{i \in N_k^0} \sum_{j \in C_2} 2c_{kj}^d \cdot y_{ij} + \pi_k^6 - \sum_{j \in C_2} \sum_{i \in N_k^0} \pi_j^7 y_{ij} + \sum_{j \in C_2} \pi_{jk}^8 \left( t_j + \sum_{i \in N_k^0} M y_{ij} \right) \quad (45)$$

$$\text{s. t. } \sum_{j \in C_2} y_{kj} = 1 \quad (46)$$

$$\sum_{j \in C_2} y_{jk'} = 1 \quad (47)$$

$$\sum_{i \in N_k^0} y_{ij} = \sum_{h \in N_k^+} y_{jh} \quad \forall j \in C_2 \quad (48)$$

$$t_j - t_i - \theta_i + M(1 - y_{ij}) \geq \tau_{ki}^d + \tau_{kj}^d \quad \forall i \in N_k^0, j \in N_k^+ \quad (49)$$

$$\sum_{i \in N_k^0} (2\tau_{kj}^d + \tau_j^d) y_{ij} \leq Q \quad \forall j \in C_2 \quad (50)$$

$$t_i \geq 0 \quad \forall i \in N_k^0 \quad (51)$$

$$y_{ij} \in \{0, 1\} \quad \forall i \in N_k^0, j \in N_k^+ \quad (52)$$

式(46)和式(47)确保无人机站  $k$  是无人机的起点和终点,与式(9)等价. 式(48)确保无人机路线的流量平衡,与式(12)等价. 式(49)计算了无人机到达顾客节点的时间. 式(50)保证了无人机一次飞行不超过其最大可持续飞行时间,与式(14)相对应. 式(51)和式(52)是变量类型约束.

## 2 分支定价算法

分支定价算法是列生成算法和分支定界算法的结合,是一种求解大规模整数规划的精确算法,可将原问题分解成多个简单的子问题,降低求解难度,提升求解效率. 本研究采用分支定价算法求解 VRP-DS,其伪代码如算法 1 所示.

算法 1 分支定价算法的伪代码

输入:  $N$ : 所有点的集合;  $T$ : 所有卡车的集合;  $D$ : 所有无人机的集合;  $K$ : 所有无人机站的集合;  
输出: 最优路线集合  $P^*$   
1: 初始化卡车路径集合  $R^t \leftarrow \emptyset$ ; 无人机路径集合  $R_k^d \leftarrow \emptyset$ ; 分支定价搜索树上活跃节点集合  $Q \leftarrow \emptyset$ ;  
2: 启发式算法获得初始解  $P^0$ , 包含可行的路径集合和每条路径的取值以及目标函数值  
3: 当前最优解  $P^c \leftarrow P^0$   
4: 使用列生成算法求解当前节点的松弛解  $P^r$   
5: 搜索树  $Q \leftarrow P^r$   
6: while  $Q \neq \emptyset$  do  
7: 从搜索树  $Q$  中取出一个解  $P$ ;  
8: if  $P$  的目标值  $\geq P^c$  的目标值 then  
9: 返回上一步,重新取出一个解  
10: if  $P$  是整数解 then  
11:  $P^c \leftarrow P$   
12: 返回上一步,重新取出一个解  
13: 按照分支规则对  $P$  分支,接着使用列生成算法解得子节点集合  $Q'(P)$   
14: 将子节点集合加入搜索树  $Q \leftarrow Q + Q'(P)$

由于 VRP-DS 包含了 VRP, 且针对 VRP 已经存在多种成熟的求解算法, 因此本研究将开源求解 OR-Tools 求解 VRP 所得到的结果, 作为 VRP-DS 的初始解.



## 2.1 列生成算法

列生成算法中主要包含线性松弛的限制主问题和子问题的求解,其伪代码如算法 2 所示.线性松弛的限制主问题模型属于线性规划可使用求解器快速求解,而子问题属于混合整数规划且被多次迭代,其求解效率对算法整体性能影响较大,因此本研究为其设计了更加高效的标号法.标号法包含三个主要概念:标签,标签的扩展和标签的占优.下文将对其进行详细阐述.

算法 2 列生成算法的伪代码

输入:  $K$ : 无人机站集合;  $R^i$ : 卡车的路径集合;  $R_k^d$ : 无人机站  $k \in K$  的无人机路径集合;  
输出: 松弛的最优路线集合  $P^s$   
1: 初始化最优子问题数量  $c = 0$ ;  
2: while  $c < 1 + |K|$  do  
3: 解线性松弛的限制主问题模型获得对偶变量  $\pi$  及当前最优解  $P^c$   
4: 使用标号法求解卡车子问题,得到检验数为负的卡车路径集合  $R^i$   
5: if  $R^i = \emptyset$  then  
6: 更新最优子问题数量  $c \leftarrow c + 1$   
7: else  
8: 更新卡车路径集合  $R^i \leftarrow R^i + R^i$   
9: for  $k \in K$  do  
10: 使用标号法求解无人机子问题,得到检验数为负的无人机路径  $R_k^d$   
11: if  $R_k^d = \emptyset$  then  
12: 更新最优子问题数量  $c \leftarrow c + 1$   
13: else  
14: 更新无人机路径集合  $R_k^d \leftarrow R_k^d + R_k^d$

### 2.1.1 求解卡车子问题

卡车子问题属于有资源限制的最短路问题<sup>[37]</sup>,本研究使用标号法进行求解,详细过程如算法 3 所示.

1) 标签: 标签中包含成本、消耗的资源信息和访问的节点等信息.在卡车子问题中使用  $L_i = (T_{rc}, T_c, T_{ct}, T_{vn}, T_{ind})$ , 表示节点  $i$  的一个标签,其中  $T_{rc}$  表示目标函数的值,即检验数;  $T_c$  表示路径当前的行驶成本;  $T_{ct}$  表示当前路径消耗的时间;  $T_{vn}$  表示当前路径已经访问的点的数量;  $T_{ind}$  是所有已经访问过的节点的集合.在程序开始前,只有一个初始标签  $L_0 = (0, 0, 0, 0, \emptyset)$ , 在仓库 0 处.

算法 3 标号法

输入:  $R$ : 卡车或无人机的路径集合;  $v_s$ : 起始点;  
 $v_e$ : 结束点;  $Succ$ : 所有节点的后继节点集合;  
输出: 最优的路径集合  $R^*$   
1: 初始化所有节点的标签集合  $L \leftarrow \emptyset$ ; 待访问节点集合  $W \leftarrow \emptyset$ ;  
起始点标签  $L[v_s]$ ;  
2: 更新待访问标签集合  $W \leftarrow L[v_s]$   
3: while  $W \neq \emptyset$  do  
4: 从待访问标签集合中取出标签  $l \in W$   
5: 获得标签  $l$  对应的节点  $v_f$   
6: for  $v_i \in Succ[v_f]$  do  
7: 从标签值  $l$  扩展得到  $j$  的标签  $l'$   
8: if  $l'$  能够占优  $v_i$  的其他标签 then  
9:  $W \leftarrow W + l'$   
10: for  $l \in L[v_e]$  do  
11: if  $l$  对应的目标函数值  $< 0$  并且对应的路径不在原有路径  $R$  中 then  
12:  $R^* \leftarrow R^* + l$  对应的路径

2) 标签的扩展: 对于节点  $i$  处的标签  $L_i = (T_{rc}^i, T_c^i, T_{ct}^i, T_{vn}^i, T_{ind}^i)$ , 扩展到节点  $j$ , 将生成标签  $L_j = (T_{rc}^j, T_c^j, T_{ct}^j, T_{vn}^j, T_{ind}^j)$ , 扩展方程如下所示

$$T_{rc}^j = T_{rc}^i + \Delta_{rc} \quad (53)$$

$$T_c^j = T_c^i + C^i \cdot d_{ij}^t \quad (54)$$

$$T_{ct}^j = T_{ct}^i + \theta_i + \tau_{ij}^t \quad (55)$$

$$T_{vn}^j = T_{vn}^i + 1 \quad (56)$$

$$T_{ind}^j = T_{ind}^i \cup \{j\} \quad (57)$$

其中  $\Delta_{rc}$  表示在从点  $i$  到点  $j$  扩展过程中, 检验数的变化量. 在扩展过程中, 若出现超过资源限制的标签如  $T_{ct}^j > l_j$ , 直接进行删除操作即可.

3) 占优规则: 若  $L_i^1 = (T_{rc}^{i1}, T_c^{i1}, T_{ct}^{i1}, T_{vn}^{i1}, T_{ind}^{i1})$  占优  $L_i^2 = (T_{rc}^{i2}, T_c^{i2}, T_{ct}^{i2}, T_{vn}^{i2}, T_{ind}^{i2})$ , 则需  $T_{rc}^{i1} \leq T_{rc}^{i2}$ ,  $T_c^{i1} \leq T_c^{i2}$ ,  $T_{ct}^{i1} \leq T_{ct}^{i2}$ ,  $T_{vn}^{i1} \leq T_{vn}^{i2}$ ,  $T_{ind}^{i1} \subset T_{ind}^{i2}$ , 且所有等号不能同时成立<sup>[37]</sup>.

### 2.1.2 求解无人机子问题

无人机子问题属于基本最短路问题 (elementary shortest path problem), 本研究采用标号法进行求解, 其求解算法框架与卡车子问题的基本一致.

1) 标签: 无人机路径的标签与卡车路径的标签相同, 点  $i$  处的标签为:  $L_i = (T_{rc}, T_c, T_{ct}, T_{vn}, T_{ind})$ , 在程序开始前, 初始标签  $L_k = (0, 0, 0, 0, \emptyset)$ , 在无人机站点  $k$  处.



2) 标签的扩展:标签的扩展方程为式(53)和式(58)、式(59)以及式(56)、式(57)所示

$$T_c^j = T_c^i + 2 \cdot C^d \cdot d_{kj}^d \quad (58)$$

$$T_{ct}^j = T_{ct}^i + \theta_i + \tau_{ki}^d + \tau_{kj}^d \quad (59)$$

3) 占优规则:由于无人机子问题与一般的最短路问题,在目标函数上有所不同,本研究为其提出了新的占优规则:若  $L_i^1 = (T_{rc}^{i1}, T_c^{i1}, T_{ct}^{i1}, T_{vn}^{i1}, T_{ind}^{i1})$  占优  $L_i^2 = (T_{rc}^{i2}, T_c^{i2}, T_{ct}^{i2}, T_{vn}^{i2}, T_{ind}^{i2})$ , 则仅需  $T_{rc}^{i1} \leq T_{rc}^{i2}, T_{ind}^{i1} \leq T_{ind}^{i2}$ , 且所有等号不能同时成立。

为了证明上述占优规则成立,需要证明若  $L_i^1$  占优  $L_i^2$  成立,则同时将这两条路从  $i$  扩展到  $j$  后,  $L_j^1$  占优  $L_j^2$  应当也成立。显然  $T_{ind}^{j1} \leq T_{ind}^{j2}$  成立,所以仅需证明  $T_{rc}^{j2} \geq T_{rc}^{j1}$  成立即可。

将无人机子问题的目标函数(45)分为两部分,其中  $RC_{kd}^1$  与  $y_{ij}$  相关的,不同路径在弧  $(i, j)$  上的变化量相同,将变化量记作  $\Delta RC_{kd}^1$ ;  $RC_{kd}^2$  与  $t_j$  相关,不同路径在弧  $(i, j)$  上的变化量不同。

$$RC_{kd} = RC_{kd}^1 + RC_{kd}^2 \quad (60)$$

$$RC_{kd}^1 = \sum_{i \in N_k^0} \sum_{j \in C_2} 2c_{kj}^d y_{ij} + \pi_k^6 - \sum_{j \in C_2} \sum_{i \in N_k^0} \pi_j^7 y_{ij} + \sum_{j \in C_2} \pi_{jk}^8 \left( \sum_{i \in N_k^0} M y_{ij} \right) \quad (61)$$

$$RC_{kd}^2 = \sum_{j \in C_2} \pi_{jk}^8 t_j \quad (62)$$

进而,可以得到下列式

$$\begin{aligned} T_{rc}^{j2} - T_{rc}^{j1} &= (T_{rc}^{i2} + \Delta RC_{kd}^1 + \pi_{jk}^8 T_{ct}^{j2}) - \\ &\quad (T_{rc}^{i1} + \Delta RC_{kd}^1 + \pi_{jk}^8 T_{ct}^{j1}) \\ &= (T_{rc}^{i2} + \Delta RC_{kd}^1 + \pi_{jk}^8 (T_{ct}^{i2} + \theta_i + \\ &\quad \tau_{ki}^d + \tau_{kj}^d)) - (T_{rc}^{i1} + \Delta RC_{kd}^1 + \pi_{jk}^8 \cdot \\ &\quad (T_{ct}^{i1} + \theta_i + \tau_{ki}^d + \tau_{kj}^d)) \\ &= (T_{rc}^{i2} - T_{rc}^{i1}) - \pi_{jk}^8 (T_{ct}^{i2} - T_{ct}^{i1}) \quad (63) \end{aligned}$$

由于无人机每次飞行仅访问一个顾客,访问完成后返回原所属无人机站点,可以得出:若  $T_{ind}^{j1} \leq T_{ind}^{j2}$ , 则  $T_{ct}^{j1} \leq T_{ct}^{j2}$  成立。由  $L_i^1$  占优  $L_i^2$  可知,  $T_{rc}^{i2} \geq T_{rc}^{i1}, \pi_{jk}^8 \geq 0$  和  $T_{ct}^{i2} \geq T_{ct}^{i1}$  成立,所以  $T_{rc}^{j2} \geq T_{rc}^{j1}$  成立,也就是  $L_j^1$  占优  $L_j^2$  成立,证明完毕。

## 2.2 分支定界算法

为了保持子问题的结构不变,本研究使用了 Boussier 和 Gendreau<sup>[38]</sup> 针对团队定向越野问题提出的基于约束的分支规则。该分支规则首先将基于路径的解转换为基于点的等价解,若存在小数解,则按照是否访问该点分为两支;否则,再将其

转换为基于弧的等价解,若仍存在小数解,则先判断弧线两端是否存在必须访问的点,若存在,则按照是否访问该弧分为两支;若不存在,则将其按照强制访问该弧线,强制访问一个点但不访问该弧,和强制不访问这个点分为三支。为了快速获得整数解,当同时存在多个候选分支节点时,本研究倾向于选择接近 1 的节点进行分支。

## 3 实验设计与结果分析

本研究所设计的算法采用 Python 3.7.3 进行编程实现,运行环境为:英特尔(R) Core(TM) 3.2 GHz 的 CPU、8GB 内存、Windows 10 系统的计算机,预定运行时间上限设置为 3 600 s。原问题模型和线性松弛的限制主问题模型均由 Gurobi 9.1.0 求解器使用默认设置运行求解。

### 3.1 实验设置

由于目前没有专门针对 VRP-DS 的测试算例,本研究将 Solomon<sup>[39]</sup> 的基准算例 C2、R2 和 RC2 转换为 VRP-DS 的测试算法,并使用  $(n, m)$  来表示算例规模,其中  $n$  表示算例中所有点的个数,  $m$  表示无人机站的个数。规模为  $(n, m)$  的算例取所有顾客点的前  $n$  个点,设置前  $m$  个点为无人机站,其余为顾客点。本研究分别生成 (10, 1)、(15, 2) 和 (19, 3) 三种规模的算例进行实验。

本研究将根据实际使用中的典型参数进行取值。每辆卡车的固定成本  $F^r$  为 20 美元<sup>[25]</sup>,每公里的运输成本  $C^r$  为 0.794 美元,无人机每公里的运输成本  $C^d$  为 0.05 美元<sup>[40]</sup>。此外,本研究将无人机的最大载荷和无人机的续航距离,转化为无人机可访问的顾客的比例,并将其设置为 80%<sup>[3]</sup>;将紧邻无人机站的前 20% 的顾客点,设置为仅能被卡车访问的顾客点的集合  $C_1$ ;将无人机的速度与卡车速度的比值设置为 2<sup>[41]</sup>。卡车的速度、最晚送达时间及其他参数与 Solomon<sup>[39]</sup> 的参数取值保持一致。

### 3.2 效率对比实验

本研究分别对三种规模的算例进行实验,比较分支定价算法与 Gurobi 的求解质量和求解效率,实验结果分别如表 2、表 3 和表 4 所示。其中, CPU(s) 表示程序所运行的总时间,若求解时间消耗完毕导致算法停止,则使用 NA 表示在给定

时间内无法求得最优解; $Obj$  表示算法求解得到的最终目标函数值;分支定价算法中的  $Gap$  表示其线性规划的结果与当前最优的整数解之间的差距,计算公式为:(整数解 - 线性解)/线性解;Gurobi 中的  $Gap$  列表示 Gurobi 计算过程中上下界之间的差距,计算公式为:(上界 - 下界)/上界; $\Delta Obj$  表示分支定价算法和 Gurobi 之间的所得结果的差距,计算公式为:(分支定价算法的  $Obj$  - Gurobi 的  $Obj$ )/Gurobi 的  $Obj$ ;  $\Delta T$  表示分支定价算法和 Gurobi 之间的计算效率的差距,计算公式为:(分支定价算法的计算时间 - Gurobi 的计算时间)/Gurobi 的计算时间。

在小型算例中,分支定价算法和 Gurobi 均可在给定的时间限制内求得最优解.但在运行效率方面,分支定价算法比商业求解器 Gurobi 要快得多,分支定价算法的最长求解时间为 0.4 s,比 Gurobi 的最短求解时间 1.2 s 还要短得多.尤其是在 RC2 型算例中,分支定价算法的求解时间均低于 0.4 s,但 Gurobi 的求解时间则在 110 s 以上.在小型算例中,在求解质量方面二者旗鼓相当,但在求解效率方面,分支定价算法表现更好。

在中型算例中,分支定价算法均可求得最优解,但 Gurobi 对于 RC2 类型的算例无一求得最优解.在算例 RC203 中, $Gap$  高达 43.48%.由前文的定义可知,表中  $Gap$  的值越小意味着解决方案的质量越好.从求解效率来看,分支定价算法耗时更短,尤其是在 RC2 类的算例中,二者的求解时间相差巨大.在算例 RC208 上,分支定价算法仅使用 1 秒即可求得最优解,而 Gurobi 在 3 600 s 内仍未求得最优解.在中型算例中,分支定价算法在求解质量和求解效率两方面均优于 Gurobi。

在大型算例中,考虑算法本身内部的  $Gap$ ,可以看到分支定价算法均求得最优解,而 Gurobi 在求解过程中的算例的  $Gap$  最高达到 44.70%.从求解效率来看,除了达到最大求解时间的算例以外,在二者都求得最优解的情况下,分支定价算法的耗时更短.在大型算例中,分支定价算法在求解质量和求解效率上均表现更优。

### 3.3 成本分析

本研究对各项成本的占比情况进行了分析,结果如图 3 所示.从该图中,可以发现总成本

中卡车的行驶成本占比高达 64%,无人机行驶成本次之,占比 17%,卡车固定成本占比 19%.若能使用无人机替代卡车完成部分顾客的访问,将能有效降低总成本。

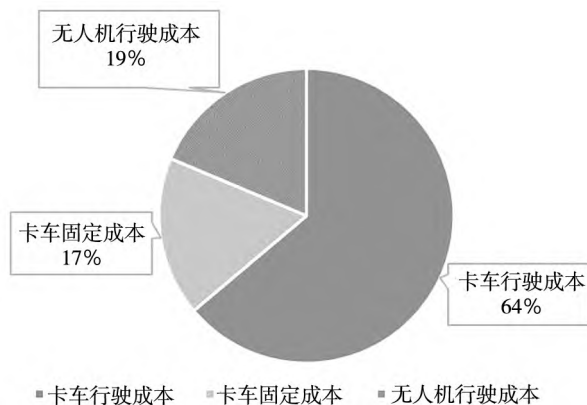


图3 各项成本占比

Fig. 3 The percentage of each cost item

### 3.4 敏感度分析

事实上,上文所提到的参数在实际配送过程中往往是不确定的.因此,本研究保持其他参数不变,针对无人机可访问的顾客比例,无人机的数量和无人机与卡车的速度比三个关键参数进行敏感度分析,希望能够为决策者提供相应管理启示。

#### 3.4.1 无人机可访问顾客比例

随着技术的进步,无人机的最大续航时间及其负载能力将会得到改善,即无人机可访问的顾客比例也会随之改变.本研究设置不同的无人机可访问的顾客比例,进行了一系列的实验,结果如图 4 所示.在三类问题中均存在目标函数值即总成本随着无人机可以访问顾客比例的增加而降低.二者呈负相关性,扩大无人机运送范围对总成本降低有显著影响。

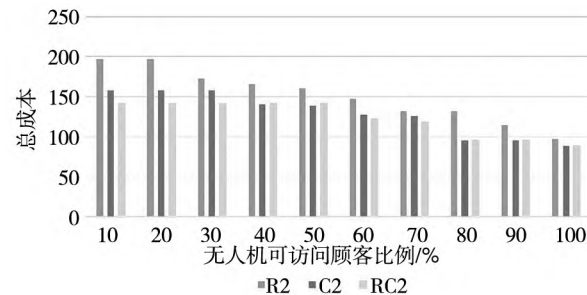


图4 无人机可访问的顾客比例对总成本的影响

Fig. 4 The impact of proportion of customers accessible by drone on the total cost

### 3.4.2 无人机的数量

本研究将无人机的数量从零架无人机增加到六架无人机,进行一系列实验,结果如图5和图6所示.在图5中,当无人机的数量从零增加到一时,三类问题的总成本都有明显下降;当从一架增加到两架时,R2和C2的总成本仍有少量的减少;再继续增加无人机,总成本均不改变.在图6中无人机访问顾客数量的增加趋势,与图5中总成本的下降趋势一致.这表明仅使用一架无人机即可满足大部分需求,使用两架及以上,存在无人机运力的浪费.在实践中,根据业务情况,购买适当数量无人机即可.

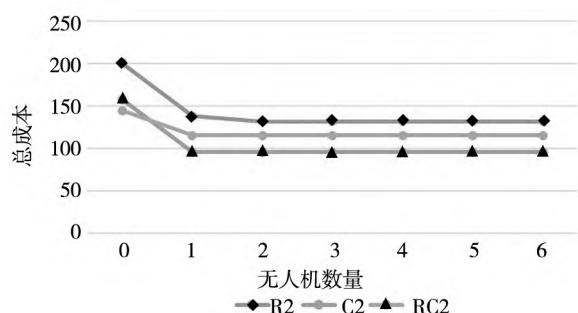


图5 无人机的数量对总成本的影响

Fig. 5 The impact of the number of drones on the total cost

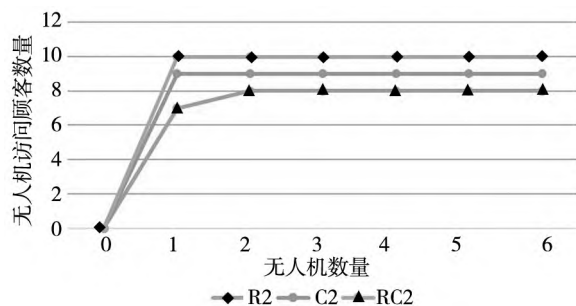


图6 无人机的数量对无人机访问的点的数量的影响

Fig. 6 The impact of the number of drones on the number of nodes visited by the drones

### 3.4.3 无人机与卡车的速度比

对于不同的系统而言,无人机与卡车的速度比往往是不确定的.本研究对两者不同的速度比进行实验,结果如图7所示.随着无人机速度的增加,时间窗对其的限制作用降低,在同一时间跨度内可访问的点更多,使得无人机访问

顾客的数量上升.同时,也存在边际效应递减的现象,无人机访问顾客数量的变化出现在无人机速度与卡车速度相等之前.在实践中,对于时间窗要求宽松的问题而言,选用速度过快的无人机对于总成本的降低没有明显帮助,选择合适的无人机速度即可.

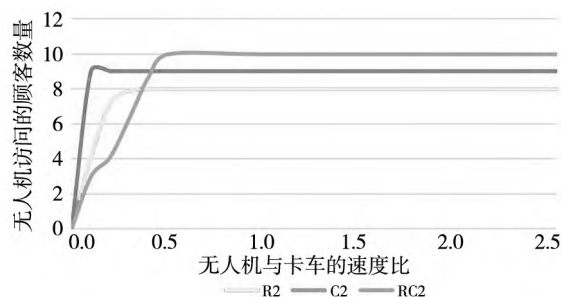


图7 无人机与卡车的速度比对无人机访问顾客数量的影响

Fig. 7 The impact of drone to truck speed ratio on the number of customers visited by drones

## 4 结束语

在物流领域,无人机站作为卡车和无人机的中介,可以辅助无人机完成各项操作,降低卡车与无人机的同步性要求,改善最后一公里问题,具有较高的应用价值.本研究提出了VRP-DS,构建了基于弧的VRP-DS原问题模型,使用Dantzig-Wolfe分解,将原问题模型分解为基于路径的限制主问题模型和卡车子问题模型以及无人机子问题模型,为了获得全局最优解,设计了精确算法-分支定价算法.

本研究将Solomon<sup>[39]</sup>的基准算例转换为VRP-DS算例,并进行一系列计算实验,实验结果验证了分支定价算法的有效性.通过成本分析发现,在总成本中卡车的行驶成本占比最高.在敏感度分析阶段发现了无人机可访问的顾客比例与总成本呈负相关,是影响总成本的关键因素,无人机的数量和速度,对总成本的影响相对较小.因此,决策者通过提高无人机负载和续航时间,增加无人机可访问的顾客比例,减少卡车的使用,可有效提升系统效益.

表 2 小型算例实验结果 ( $n = 10, m = 1$ )  
Table 2 Experimental results of small-scale instances ( $n = 10, m = 1$ )

问题类型	算例名称	分支定价			Gurobi			$\Delta Obj$	$\Delta T$
		$Obj$	$CPU(s)$	$Gap$	$Obj$	$CPU(s)$	$Gap$		
R2	R201	102.46	0.1	0.00%	102.46	1.3	0.00%	0.00%	-92.35%
	R202	102.46	0.2	0.00%	102.46	1.2	0.00%	0.00%	-85.00%
	R203	102.46	0.2	0.00%	102.46	1.2	0.00%	0.00%	-86.57%
	R204	102.46	0.1	0.00%	102.46	1.3	0.00%	0.00%	-89.81%
	R205	102.46	0.1	0.00%	102.46	1.4	0.00%	0.00%	-94.10%
	R206	102.46	0.2	0.00%	102.46	1.4	0.00%	0.00%	-86.65%
	R207	102.46	0.2	0.00%	102.46	1.4	0.00%	0.00%	-86.02%
	R208	102.46	0.1	0.00%	102.46	1.5	0.00%	0.00%	-91.48%
	R209	102.46	0.1	0.00%	102.46	1.7	0.00%	0.00%	-95.91%
	R210	102.46	0.2	0.00%	102.46	1.4	0.00%	0.00%	-85.87%
	R211	102.46	0.1	0.00%	102.46	1.3	0.00%	0.00%	-93.30%
C2	C201	87.47	0.2	0.00%	87.47	2.0	0.00%	0.00%	-89.47%
	C202	87.47	0.1	0.00%	87.47	9.3	0.00%	0.00%	-98.79%
	C203	87.47	0.1	0.00%	87.47	9.3	0.00%	0.00%	-98.77%
	C204	87.47	0.1	0.00%	87.47	10.2	0.00%	0.00%	-98.90%
	C205	87.47	0.1	0.00%	87.47	5.1	0.00%	0.00%	-97.74%
	C206	87.47	0.1	0.00%	87.47	7.6	0.00%	0.00%	-98.51%
	C207	87.47	0.4	0.00%	87.47	7.5	0.00%	0.00%	-95.18%
	C208	87.47	0.1	0.00%	87.47	6.7	0.00%	0.00%	-98.32%
RC2	RC201	99.51	0.2	0.00%	99.51	78.7	0.00%	0.00%	-99.70%
	RC202	99.51	0.3	0.00%	99.51	109.2	0.00%	0.00%	-99.71%
	RC203	99.51	0.3	0.00%	99.51	107.3	0.00%	0.00%	-99.70%
	RC204	99.51	0.3	0.00%	99.51	91.5	0.00%	0.00%	-99.65%
	RC205	99.51	0.3	0.00%	99.51	104.1	0.00%	0.00%	-99.70%
	RC206	99.51	0.2	0.00%	99.51	95.8	0.00%	0.00%	-99.74%
	RC207	99.51	0.2	0.00%	99.51	109.7	0.00%	0.00%	-99.78%
	RC208	99.51	0.2	0.00%	99.51	106.3	0.00%	0.00%	-99.86%

表 3 中型算例实验结果 ( $n = 15, m = 2$ )  
Table 3 Experimental results of medium-scale instances ( $n = 15, m = 2$ )

问题类型	算例名称	分支定价			Gurobi			$\Delta Obj$	$\Delta T$
		$Obj$	$CPU(s)$	$Gap$	$Obj$	$CPU(s)$	$Gap$		
R2	R201	132.01	0.2	0.00%	132.01	35.3	0.00%	0.00%	-99.54%
	R202	132.01	0.2	0.00%	132.01	36.5	0.00%	0.00%	-99.36%
	R203	132.01	0.2	0.00%	132.01	20.4	0.00%	0.00%	-98.99%
	R204	132.01	0.2	0.00%	132.01	33.1	0.00%	0.00%	-99.46%
	R205	132.01	0.1	0.00%	132.01	34.0	0.00%	0.00%	-99.56%
	R206	132.01	0.2	0.00%	132.01	21.8	0.00%	0.00%	-98.98%
	R207	132.01	0.2	0.00%	132.01	34.0	0.00%	0.00%	-99.35%
	R208	132.01	0.2	0.00%	132.01	26.1	0.00%	0.00%	-99.35%
	R209	132.01	0.2	0.00%	132.01	39.3	0.00%	0.00%	-99.60%
	R210	132.01	0.6	0.00%	132.01	42.8	0.00%	0.00%	-98.60%
	R211	132.01	0.2	0.00%	132.01	18.2	0.00%	0.00%	-98.93%



续表 3  
Table 3 Continues

问题类型	算例名称	分支定价			Gurobi			$\Delta Obj$	$\Delta T$
		$Obj$	$CPU(s)$	$Gap$	$Obj$	$CPU(s)$	$Gap$		
C2	C201	95.46	18.0	0.00%	95.46	19.6	0.00%	0.00%	-8.50%
	C202	95.46	0.3	0.00%	95.46	298.4	0.00%	0.00%	-99.91%
	C203	95.46	0.3	0.00%	95.46	170.9	0.00%	0.00%	-99.84%
	C204	95.46	0.3	0.00%	95.46	184.7	0.00%	0.00%	-99.83%
	C205	95.46	0.3	0.00%	95.46	146.0	0.00%	0.00%	-99.79%
	C206	95.46	0.3	0.00%	95.46	135.4	0.00%	0.00%	-99.77%
	C207	95.46	0.2	0.00%	95.46	95.5	0.00%	0.00%	-99.75%
	C208	95.46	0.2	0.00%	95.46	173.1	0.00%	0.00%	-99.87%
RC2	RC201	115.32	0.5	0.00%	115.32	NA	16.79%	0.00%	
	RC202	115.32	10.8	0.00%	115.32	NA	16.91%	0.00%	
	RC203	115.32	3.3	0.00%	115.32	NA	43.48%	0.00%	
	RC204	115.32	3.4	0.00%	115.32	NA	37.92%	0.00%	
	RC205	115.32	4.8	0.00%	115.32	NA	12.94%	0.00%	
	RC206	115.32	7.0	0.00%	115.32	NA	12.92%	0.00%	
	RC207	115.32	0.8	0.00%	115.32	NA	26.11%	0.00%	
	RC208	115.32	0.8	0.00%	115.32	NA	21.79%	0.00%	

表 4 大型算例实验结果 ( $n = 19, m = 3$ )  
Table 4 Experimental results of large-scale instances ( $n = 19, m = 3$ )

问题类型	算例名称	分支定价			Gurobi			$\Delta Obj$	$\Delta T$
		$Obj$	$CPU(s)$	$Gap$	$Obj$	$CPU(s)$	$Gap$		
R2	R201	154.47	276.5	0.00%	154.47	405.9	0.00%	0.00%	-31.87%
	R202	154.47	47.0	0.00%	154.47	926.7	0.00%	0.00%	-94.93%
	R203	154.47	51.0	0.00%	154.47	2 083.6	0.00%	0.00%	-97.55%
	R204	154.47	53.3	0.00%	154.47	NA	9.16%	0.00%	
	R205	154.47	53.7	0.00%	154.47	288.6	0.00%	0.00%	-81.41%
	R206	154.47	60.5	0.00%	154.47	3 264.9	0.00%	0.00%	-98.15%
	R207	154.47	55.9	0.00%	154.47	1 084.4	0.00%	0.00%	-94.85%
	R208	154.47	53.5	0.00%	154.47	318.4	0.00%	0.00%	-83.20%
	R209	154.47	55.1	0.00%	154.47	216.9	0.00%	0.00%	-74.62%
	R210	154.47	48.7	0.00%	154.47	3 149.9	0.00%	0.00%	-98.45%
	R211	154.47	57.2	0.00%	154.47	783.3	0.00%	0.00%	-92.69%
C2	C201	138.75	209.9	0.00%	138.75	225.7	0.00%	0.00%	-7.04%
	C202	137.40	176.3	0.00%	137.40	1 546.9	0.00%	0.00%	-88.60%
	C203	137.40	615.4	0.00%	137.40	2 100.7	0.00%	0.00%	-70.71%
	C204	137.40	366.6	0.00%	137.40	2 167.2	0.00%	0.00%	-83.08%
	C205	137.40	202.3	0.00%	137.40	389.5	0.00%	0.00%	-48.06%
	C206	137.40	204.7	0.00%	137.40	476.8	0.00%	0.00%	-57.06%
	C207	138.75	71.3	0.00%	138.75	170.1	0.00%	0.00%	-58.11%
	C208	137.40	75.1	0.00%	137.40	795.1	0.00%	0.00%	-90.55%
RC2	RC201	143.07	2 643.9	0.00%	143.07	NA	44.70	-0.91%	
	RC202	143.07	655.5	0.00%	143.07	NA	41.45	0.00%	
	RC203	143.07	790.7	0.00%	143.07	NA	40.97	0.00%	
	RC204	143.07	887.0	0.00%	143.07	NA	41.96	0.00%	
	RC205	143.07	327.6	0.00%	143.07	NA	40.54	0.00%	
	RC206	143.07	128.5	0.00%	143.07	NA	40.53	0.00%	
	RC207	143.07	377.4	0.00%	143.07	NA	40.37	0.00%	
	RC208	143.07	681.5	0.00%	143.07	NA	39.82	0.00%	

## 参 考 文 献:

- [1] 张源凯, 胡祥培, 黄敏芳, 等. 网上超市拆分订单合并打包策略经济决策模型[J]. 管理科学学报, 2019, 22(10): 24–36.  
Zhang Yuankai, Hu Xiangpei, Huang Minfang, et al. Economic decision model for package consolidation in fulfilling split orders of online supermarkets[J]. Journal of Management Sciences in China, 2019, 22(10): 24–36. (in Chinese)
- [2] 池毛毛, 赵 晶, 李延晖, 等. 电子商务平台吸附能力的影响机制研究——平台柔性和控制机制的交互效应[J]. 管理科学学报, 2018, 21(7): 35–51.  
Chi Maomao, Zhao Jing, Li Yanhui, et al. Effect of e-business platform attractive capabilities: The interaction of effects of platform flexibility and control mechanisms[J]. Journal of Management Sciences in China, 2018, 21(7): 35–51. (in Chinese)
- [3] Otto A, Agatz N, Campbell J, et al. Optimization approaches for civil applications of unmanned aerial vehicles(UAVs) or aerial drones: A survey[J]. Networks, 2018, 72(4): 411–458.
- [4] Perboli G, Rosano M. Parcel delivery in urban areas: Opportunities and threats for the mix of traditional and green business models[J]. Transportation Research Part C: Emerging Technologies, 2019, 99: 19–36.
- [5] 饶卫振, 朱庆华, 金 淳, 等. 协作车辆路径成本分摊问题的 B-T Shapley 方法[J]. 管理科学学报, 2019, 22(1): 107–126.  
Rao Weizhen, Zhu Qinghua, Jin Chun, et al. A Binary Tree Shapely method for cost sharing of the collaborative vehicle routing problem[J]. Journal of Management Sciences in China, 2019, 22(1): 107–126. (in Chinese)
- [6] 周鲜成, 刘长石, 周开军, 等. 时间依赖型绿色车辆路径模型及改进蚁群算法[J]. 管理科学学报, 2019, 22(5): 57–68.  
Zhou Xiancheng, Liu Changshi, Zhou Kaijun, et al. Improved ant colony algorithm and modelling of time-dependent green vehicle routing problem[J]. Journal of Management Sciences in China, 2019, 22(5): 57–68. (in Chinese)
- [7] 徐小峰, 姜明月, 邓忆瑞. 整合逆向物流协同配送动态路径优化问题研究[J]. 管理科学学报, 2021, 24(10): 106–126.  
Xu Xiaofeng, Jiang Mingyue, Deng Yirui. Dynamic vehicle routing problem with simultaneous pickup and delivery in collaborative distribution under demand concurrent[J]. Journal of Management Sciences in China, 2021, 24(10): 106–126. (in Chinese)
- [8] 环球网. 顺丰无人机快递接驳柜面市[EB/OL]. <https://uav.huanqiu.com/article/9CaKrnKeTKU>. 2018.  
World Wide Web. SF Express Drone Connection Counter Market[EB/OL]. <https://uav.huanqiu.com/article/9CaKrnKeTKU>. 2018. (in Chinese)
- [9] 张 维, 杨彦武, 乔 红, 等. 混合智能管理系统理论与方法研究[J]. 管理科学学报, 2021, 24(8): 10–17.  
Zhang Wei, Yang Yanwu, Qiao Hong, et al. Hybrid intelligence management system research: Theory and methods[J]. Journal of Management Sciences in China, 2021, 24(8): 10–17. (in Chinese)
- [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] Carlsson J G, Song S. Coordinated logistics with a truck and a drone[J]. Management Science, 2018, 64(9): 4052–4069.
- [13] 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.

- [14] 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.
- [15] Jeong H Y, Song B D, Lee S. Truck-drone hybrid delivery routing: Payload-energy dependency and No-Fly zone[J]. *International Journal of Production Economics*, 2019, 214: 220 – 233.
- [16] Bouman P, Agatz N, Schmidt M. Dynamic programming approaches for the traveling salesman problem with drone[J]. *Networks*, 2018, 72(4): 528 – 542.
- [17] Roberti R, Ruthmair M. Exact methods for the traveling salesman problem with drone[J]. *Transportation Science*, 2021, 55(2): 315 – 335.
- [18] de Freitas J C, Penna P H V. A variable neighborhood search for flying sidekick traveling salesman problem[J]. *International Transactions in Operational Research*, 2020, 27(1): 267 – 290.
- [19] Ha Q M, Deville Y, Pham Q D, et al. On the min-cost traveling salesman problem with drone[J]. *Transportation Research Part C: Emerging Technologies*, 2018, 86: 597 – 621.
- [20] 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.
- [21] Tu P A, Dat N T, Dung P Q. Traveling Salesman Problem with Multiple Drones[C]. In *Proceedings of the Ninth International Symposium on Information and Communication Technology*, 2018: 46 – 53.
- [22] Wang X, Poikonen S, Golden B. The vehicle routing problem with drones; Several worst-case results[J]. *Optimization Letters*, 2017, 11(4): 679 – 697.
- [23] Poikonen S, Wang X, Golden B. The vehicle routing problem with drones; Extended models and connections[J]. *Networks*, 2017, 70(1): 34 – 43.
- [24] Schermer D, Moeini M, Wendt O. A matheuristic for the vehicle routing problem with drones and its variants[J]. *Transportation Research Part C: Emerging Technologies*, 2019, 106: 166 – 204.
- [25] Wang Z, Sheu J B. Vehicle routing problem with drones[J]. *Transportation Research Part B: Methodological*, 2019, 122: 350 – 364.
- [26] Schermer D, Moeini M, Wendt O. A hybrid VNS/Tabu search algorithm for solving the vehicle routing problem with drones and en route operations[J]. *Computers & Operations Research*, 2019, 109: 134 – 158.
- [27] Karak A, Abdelghany K F. The hybrid vehicle-drone routing problem for pick-up and delivery services[J]. *Transportation Research Part C: Emerging Technologies*, 2019, 102: 427 – 449.
- [28] Sacramento D, Pisinger D, Ropke S. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones[J]. *Transportation Research Part C: Emerging Technologies*, 2019, 102: 289 – 315.
- [29] Tamke F, Buscher U. A branch-and-cut algorithm for the vehicle routing problem with drones[J]. *Transportation Research Part B Methodological*, 2021, 144(4): 174 – 203.
- [30] Baloch G, Gzara F. Strategic network design for parcel delivery with drones under competition[J]. *Transportation Science*, 2020, 54(1): 204 – 228.
- [31] Mbiadou S, Laurent D, Dominique F, et al. An iterative two-step heuristic for the parallel drone scheduling traveling salesman problem[J]. *Networks*, 2018, 72(4): 459 – 474.
- [32] Dell'Amico M, Montemanni R, Novellani S. Matheuristic algorithms for the parallel drone scheduling traveling salesman problem[J]. *Annals of Operations Research*, 2020, 289(2): 211 – 226.
- [33] Kim S, Moon I. Traveling salesman problem with a drone station[J]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2018, 49(1): 42 – 52.
- [34] Li Y, Zhang G, Pang Z, et al. Continuum approximation models for joint delivery systems using trucks and drones[J]. *Enterprise Information Systems*, 2020, 14(4): 406 – 435.
- [35] Schermer D, Moeini M, Wendt O. The Traveling Salesman Drone Station Location Problem[C]. In *World Congress on*

- Global Optimization, 2019: 1129 – 1138.
- [36] Lenstra J K, Kan A R. Complexity of vehicle routing and scheduling problems[J]. Networks, 1981, 11(2): 221 – 227.
- [37] Lozano L, Duque D, Medaglia A L. An exact algorithm for the elementary shortest path problem with resource constraints [J]. Transportation Science, 2016, 50(1): 348 – 357.
- [38] Boussier S, Feillet D, Gendreau M. An exact algorithm for team orienteering problems[J]. 4OR, 2007, 5(3): 211 – 230.
- [39] Solomon M M. Algorithms for the vehicle routing and scheduling problems with time window constraints[J]. Operations Research, 1987, 35(2): 254 – 265.
- [40] Liu Y, Liu Z, Shi J, et al. Two-echelon routing problem for parcel delivery by cooperated truck and drone[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2021, 51(12): 7450 – 7465.
- [41] Ferrandez S M, Harbison T, Weber T, et al. Optimization of a truck-drone in tandem delivery network using k-means and genetic algorithm[J]. Journal of Industrial Engineering and Management, 2016, 9(2): 374 – 388.

## The vehicle routing problem with drone stations

*FANG Kan<sup>1, 2</sup>, AN Yi-dan<sup>1</sup>, ZHU Ning<sup>1</sup>, HUANG Dian<sup>1, 2, 3\*</sup>*

1. College of Management and Economics, Tianjin University, Tianjin 300072, China;
2. Laboratory of Computation and Analytics of Complex Management Systems, Tianjin University, Tianjin 300072, China;
3. School of New Media and Communication, Tianjin University, Tianjin 300072, China

**Abstract:** In recent years, drone scheduling has received increasing attention from researchers. In this study, a vehicle routing problem with drone stations is proposed, and an arc-based optimization model is developed with an objective of minimizing the total cost, considering various constraints such as the latest arrival time of packages, the limited number of drones and the maximum flying duration of drones. Using the Dantzig-Wolfe decomposition, the arc-based model is decomposed into the path-based main problem model, the truck sub-problem model, and the drone sub-problem model. A branch and price algorithm is designed to solve the problem and to obtain the global optimal solution. Extensive numerical experimental results show that our proposed branch and price algorithm outperforms the main stream commercial solver in terms of solution time and solution quality. In addition, the truck travel cost accounts for a high proportion of the total cost. Moreover, compared with the number of drones and the speed ratio of drones to trucks, the proportion of customers accessible by drones has a more significant impact on the total cost. Therefore, expanding the applicability of drones can effectively reduce the total cost.

**Key words:** drone stations; drone scheduling; vehicle routing problem; branch and price algorithm