

Contents

| | |
|--|-----------|
| 1 TDRP: Truck-Drone collaborative Routing Problem | 1 |
| 1 Traveling Salesman Problem with Drone | 3 |
| 1.1 Flying Sidekick Traveling Salesman Problem | 3 |
| 1.1.1 Flying Sidekick Traveling Salesman Problem with Multiple Drops | 8 |
| 1.2 Parallel Drone Scheduling Traveling Salesman Problem | 14 |
| References | 19 |

Part

1

*TDRP: Truck-Drone collaborative
Routing Problem*

1

Traveling Salesman Problem with Drone

1.1 Flying Sidekick Traveling Salesman Problem

Flying Sidekick Traveling Salesman Problem (FSTSP) 由 Murray(2015) 等^[1] 提出。FSTSP 描述：假定有 c 个顾客需要服务，这个服务可以由无人机或者卡车来提供，但是有些顾客由于某些限制（比如包裹重量超过无人机的载重限制）只能由卡车提供服务。卡车和无人机必须从单一仓库出发，并且返回该仓库一次，即不能重复访问该仓库，无人机和卡车可以同时或者分别离开（返回）仓库。在无人机起飞前需要卡车司机装载包裹和更换电池的时间 s_L ，降落时需要给无人机装卸货物和充电的时间 s_R 。每次无人机的一次服务称为一次 sortie，一次 sortie 分为三个节点 $\langle i, j, k \rangle$ ，起飞点 i 可以是仓库也可以是顾客点，中间节点 j 是需要服务的顾客节点，节点 k 可以是仓库也可以是卡车所在的顾客节点，无人机在整个运输过程中可以进行多次 sortie 服务多个顾客，但是一次 sortie 的时间必须在无人机的续航时间内。FSTSP 的目标是最小化服务所有顾客并且返回仓库（无人机和卡车都返回）的时间。

关键假设如下：

- 无人机每次 sortie 的过程中只能服务一个顾客节点，但是在这期间卡车可以服务多个顾客节点。
- 无人机被假定为匀速飞行，如果无人机比卡车提前到达会合点则无人机不能在中途停下休息以节省电量。
- 无人机可以在降落点重新发射，但是无人机不能返回上一次的发射点。
- 无人机和卡车的会合点必须在顾客节点，而不能在中间的任何位置会合，并且卡车不会重新访问已经服务过的顾客节点来回收无人机。
- 无人机和卡车都不能访问除了仓库以外的非顾客节点（即只考虑简化过后的实际情况），并且无人机和卡车都不能重新访问已经服务过的顾客节点。
- 如果无人机返回仓库则不能再次进行服务，这是基于大多顾客节点都离仓库较远（大于无人机的续航里程）的假设，在无人机可以直接起飞服务顾客节点的假设下，PDSTSP 会更加适合。

FSTSP 数学模型的符号含义如表1-1所示。

FSTSP 数学模型可以表示为 MILP 1.1。

表 1-1: FSTSP 模型符号及含义

| 符号 | 含义 |
|--|--|
| 0 | 起点仓库 |
| $c + 1$ | 终点仓库（和起点仓库相同，只是为了建模方便的另一个记号） |
| $C = \{1, 2, \dots, c\}$ | 全部客户集合 |
| $C' \subseteq C$ | 无人机可访问的客户集合 |
| $N_0 = \{0, 1, 2, \dots, c\}$ | 流出节点集合 |
| $N_+ = \{1, 2, \dots, c + 1\}$ | 流入节点集合 |
| $N = \{0, 1, 2, \dots, c, c + 1\}$ | 全部节点集合 |
| $\langle i, j, k \rangle \in P, i \in N_0, j \in \{C' : j \neq i\},$ $k \in \{N_+ : k \neq i, k \neq j, \tau'_{ij} + \tau'_{jk} \leq e\}$ | 无人机飞行路径集合 |
| $\tau'_{ij}/\tau_{ij}, i \in N_0, j \in N_+, i \neq j, \tau_{0,c+1} \equiv 0^{\text{a}}$ | 弧 $\langle i, j \rangle$ 的飞行/行驶时间成本 |
| s_L/s_R | 无人机发射/回收耗时 |
| e | 无人机续航时长，以单位时间来衡量 |
| $x_{ij} \in \{0, 1\}, i \in N_0, j \in N_+, i \neq j$ | 卡车路由决策变量 |
| $y_{ijk} \in \{0, 1\}, i \in N_0, j \in C, k \in \{N_+ : \langle i, j, k \rangle \in P\}$ | 无人机路由决策变量 |
| $t'_i/t_i \geq 0, i \in N_+, t'_0 = t_0 = 0$ | 无人机/卡车有效到达时间戳辅助变量 |
| $p_{ij} \in \{0, 1\}^{\text{b}}, p_{0j} = 1, \forall j \in C$ | 卡车访问次序先后辅助变量（为了确保无人机连续的 sortie 和卡车访问的顺序一致 ^c ） |
| $1 \leq u_i \leq c + 2, i \in N_+$ | 卡车破子圈辅助变量（和标准 TSP 的 MTZ 形式破子圈辅助变量类似 ^d ） |

^a 出于完备性的考虑，当只有一个顾客节点的时候，这个顾客将由无人机从仓库直接起飞进行服务。^b 当卡车访问顾客节点 $j \in \{C : j \neq i\}$ 时，顾客节点 $i \in C$ 已经在之前的某个时间点被卡车访问过了，则 $p_{ij} = 1$ 。^c 当顾客节点 i 或者 j 仅被无人机服务时， p_{ij} 的取值就不重要。^d u_i 表示顾客点 i 在卡车访问的路径中的次序，比如 $u_5 = 1$ 表示顾客点 $i = 5$ 是卡车访问路径中的第 1 个节点，但是不同于 TSP，在 FSTSP 中需要通过约束将无人机服务的顾客点 i 排除在外。

Model 1.1: FSTSP MILP

$$\min t_{c+1} \quad (1-1)$$

$$\text{s.t. } \sum_{\substack{i \in N_0 \\ i \neq j}} x_{ij} + \sum_{\substack{i \in N_0 \\ i \neq j}} \sum_{\substack{k \in N_+ \\ \langle i, j, k \rangle \in P}} y_{ijk} = 1, \quad \forall j \in C \quad (1-2)$$

$$\sum_{j \in N_+} x_{0j} = 1 \quad (1-3)$$

$$\sum_{i \in N_0} x_{i,c+1} = 1 \quad (1-4)$$

$$u_i - u_j + 1 \leq (c+2)(1 - x_{ij}), \quad \forall i \in C, j \in \{N_+ : j \neq i\} \quad (1-5)$$

$$\sum_{\substack{i \in N_0 \\ i \neq j}} x_{ij} = \sum_{\substack{k \in N_+ \\ k \neq j}} x_{jk}, \quad \forall j \in C \quad (1-6)$$

$$\sum_{\substack{j \in C \\ j \neq i}} \sum_{\substack{k \in N_+ \\ \langle i, j, k \rangle \in P}} y_{ijk} \leq 1, \quad \forall i \in N_0 \quad (1-7)$$

$$\sum_{\substack{i \in N_0 \\ i \neq k}} \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P}} y_{ijk} \leq 1, \quad \forall k \in N_+ \quad (1-8)$$

$$2y_{ijk} \leq \sum_{\substack{h \in N_0 \\ h \neq i}} x_{hi} + \sum_{\substack{l \in C \\ l \neq k}} x_{lk}, \quad \forall i \in C, j \in \{C : j \neq i\}, k \in \{N_+ : \langle i, j, k \rangle \in P\} \quad (1-9)$$

$$y_{0jk} \leq \sum_{\substack{h \in N_0 \\ h \neq k}} x_{hk}, \quad \forall j \in C, k \in \{N_+ : \langle 0, j, k \rangle \in P\} \quad (1-10)$$

$$u_k - u_i \geq 1 - (c+2) \left(1 - \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall i \in C, k \in \{N_+ : k \neq i\} \quad (1-11)$$

$$t'_i \geq t_i - M \left(1 - \sum_{\substack{j \in C \\ j \neq i}} \sum_{\substack{k \in N_+ \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall i \in C \quad (1-12)$$

$$t'_i \leq t_i + M \left(1 - \sum_{\substack{j \in C \\ j \neq i}} \sum_{\substack{k \in N_+ \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall i \in C \quad (1-13)$$

$$t'_k \geq t_k - M \left(1 - \sum_{\substack{i \in N_0 \\ i \neq k}} \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall k \in N_+ \quad (1-14)$$

$$t'_k \leq t_k + M \left(1 - \sum_{\substack{i \in N_0 \\ i \neq k}} \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall k \in N_+ \quad (1-15)$$

continued

$$t_k \geq t_h + \tau_{hk} + s_L \left(\sum_{\substack{l \in C \\ l \neq k}} \sum_{\substack{m \in N_+ \\ \langle k, l, m \rangle \in P}} y_{klm} \right) + s_R \left(\sum_{\substack{i \in N_0 \\ i \neq k}} \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P}} y_{ijk} \right) - M(1 - x_{hk}),$$

$$\forall h \in N_0, k \in \{N_+ : k \neq h\}$$
(1-16)

$$t'_j \geq t'_i + \tau'_{ij} - M \left(1 - \sum_{\substack{k \in N_+ \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall j \in C', i \in \{N_0 : i \neq j\}$$
(1-17)

$$t'_k \geq t'_j + \tau'_{jk} + s_R - M \left(1 - \sum_{\substack{i \in N_0 \\ \langle i, j, k \rangle \in P}} y_{ijk} \right), \quad \forall j \in C', k \in \{N_+ : k \neq j\}$$
(1-18)

$$t'_k - (t'_j - \tau'_{ij}) \leq e + M(1 - y_{ijk}), \quad \forall k \in N_+, j \in \{C : j \neq k\}, i \in \{N_0 : \langle i, j, k \rangle \in P\}$$
(1-19)

$$u_i - u_j \geq 1 - (c + 2)p_{ij}, \quad \forall i \in C, j \in \{C : j \neq i\}$$
(1-20)

$$u_i - u_j \leq -1 + (c + 2)(1 - p_{ij}), \quad \forall i \in C, j \in \{C : j \neq i\}$$
(1-21)

$$p_{ij} + p_{ji} = 1, \quad \forall i \in C, j \in \{C : j \neq i\}$$
(1-22)

$$t'_l \geq t'_k - M \left(3 - \sum_{\substack{j \in C \\ \langle i, j, k \rangle \in P \\ j \neq l}} y_{ijk} - \sum_{\substack{m \in C \\ m \neq i \\ m \neq k \\ m \neq l}} \sum_{\substack{n \in N_+ \\ \langle l, m, n \rangle \in P \\ n \neq i \\ n \neq k}} y_{lmn} - p_{il} \right)$$
(1-23)

$$\forall i \in N_0, k \in \{N_+ : k \neq i\}, l \in \{C : l \neq i, l \neq k\}$$

$$t_0 = 0$$
(1-24)

$$t'_0 = 0$$
(1-25)

$$p_{0j} = 1, \quad \forall j \in C$$
(1-26)

$$x_{ij} \in \{0, 1\}, \quad \forall i \in N_0, j \in \{N_+ : j \neq i\}$$
(1-27)

$$y_{ijk} \in \{0, 1\}, \quad \forall i \in N_0, j \in \{C : j \neq i\}, k \in \{N_+ : \langle i, j, k \rangle \in P\}$$
(1-28)

$$1 \leq u_i \leq c + 2, \quad \forall i \in N_+$$
(1-29)

$$t_i \geq 0, \quad \forall i \in N$$
(1-30)

$$t'_i \geq 0, \quad \forall i \in N$$
(1-31)

$$p_{ij} \in \{0, 1\}, \quad \forall i \in N_0, j \in \{C : j \neq i\}$$
(1-32)

目标函数 1-1 追求最小化卡车到达终点仓库 $c + 1$ 的有效时间 t_{c+1} , 通过约束 1-14 和 1-15 来对齐无人机和卡车最晚到达终点仓库的时间, 所以目标函数相当于 $\min\{\max\{t_{c+1}, t'_{c+1}\}\}$ 。

约束条件可以分为四类^[2]:

- 客户有关的约束: 约束 1-2 要求对于任何一位顾客 j , 必须且只能被卡车 (或无人机) 服务一次。
- 卡车有关的约束:

- 卡车流平衡约束：约束 1-3 要求卡车必须从起点仓库流出一次，约束 1-4 要求卡车必须从终点仓库流入一次，约束 1-6 要求卡车在中间节点满足流入和流出相等的流平衡约束。

- 卡车破子圈约束：约束 1-5 是 MTZ 形式的破子圈约束^[3-4]，去除了子圈存在的可能，这里 M 取到了 $u_i - u_j + 1$ 的上界 $c + 2$ ， u_i 可以理解为点 i 的访问次序，比如 $u_1 = 5$ 可以理解为点 1 是从出发点开始，第五个被访问到的点。

- 无人机有关的约束：

- 无人机发射、回收节点流约束：约束 1-7 表示无人机可以从非终点仓库流出，约束 1-8 表示无人机可以从非起点仓库流入。

- 无人机访问、回收节点时间戳约束：约束 1-17 表示无人机访问顾客的时间戳应该符合时间逻辑，即不早于起飞时间戳 $t'_i +$ 前往服务顾客点的飞行时长 τ'_{ij}^1 ，约束 1-18 表示无人机回到卡车的时间戳应该符合时间逻辑，即不早于访问顾客点的 $t'_j +$ 返回卡车的飞行时长 $\tau'_{jk} +$ 回收无人机用时 s_R^2 。

- 无人机电量续航约束：约束 1-19 表示无人机的飞行时间不能超过其续航时间，即到达会合点 t'_k 的有效时间 – 无人机从节点 i 的起飞时间 $(t'_j - \tau'_{ij})$ （不直接使用 t'_i 是因为 t'_i 不是起飞的时间戳而是无人机到达 i 点的时间戳）要在无人机的续航时间 e 之内。

- 无人机飞行次序约束：约束 1-23 要求无人机对于任意两条路径 $\langle i, j, k \rangle$ 和 $\langle l, m, n \rangle$ 而言，如果无人机先从节点 i 起飞之后的某个时间才从节点 l ($p_{il} = 1$) 起飞，则无人机必须先完成上一次飞行才能继续下一次飞行 ($t'_l \geq t'_k$)，并且任意两条路径之间无交叉。

- 无人机和卡车同步有关的约束：

- 无人机发射、回收点卡车访问约束：约束 1-9 要求对于非起点发射的无人机 ($\forall i \in C$)，卡车必须经过无人机的起飞点 i 和降落点 k ，约束 1-10 要求对于从起点仓库起飞的无人机来说，卡车必须经过无人机的降落点。

- 无人机访问顾客时卡车访问次序约束：约束 1-11 要求卡车必须先访问无人机的起飞点再访问无人机的降落点。

- 无人机发射点时间戳约束：约束 1-12 和 1-13 为无人机发射点的有效时间约束，要求无人机在发射节点的有效时间等于卡车在该点的有效时间，共同实现了卡车和无人机在发射节点时间上的对齐。

- 无人机回收点时间戳约束：约束 1-14 和 1-15 为无人机回收点的有效时间约束，要求无人机在回收节点的有效时间等于卡车在该点的有效时间，共同实现了卡车和无人机在回收节点时间上的对齐。

- 卡车访问顾客节点时间戳约束：约束 1-16 要求卡车访问当前顾客点 k 时必须要先将需要起飞的无人机 s_L 发射或者需要降落的无人机 s_R 回收，并且要大于到达顾客点 h 的有效时间戳 $t_h +$ 路径 $\langle h, k \rangle$ 所花费的时间 τ_{hk}^3 。

- 辅助变量和决策变量：

¹起飞时间 s_L 没有被包含进来的原因是，当无人机从顾客点 i 起飞时，约束 1-12 和 1-13 会使得 $t'_i = t_i$ ，然后约束 1-14 和 1-15 会将起飞的时间 s_L 包含在飞行到顾客点 j 的时间内。

²这里降落的时间 s_R 必须被包含进来的原因是，卡车可能比无人机更快到达会合点 k （原文中举出了一个例子以助于理解^[1]）。

³这里假设卡车从 $h \in N_0$ 行驶到 $k \in N_+$ 。

- 卡车访问次序约束：约束1-20, 1-21和1-22决定了卡车访问次序 p_{ij} 取值的合理性， u_i 和 p_{ij} 主要用于约束被卡车访问的节点之间的次序，对于被无人机服务的顾客点 i 或者 j 来说， u_i 和 p_{ij} 的取值并不重要。
- 辅助变量及决策变量的初始值和取值范围：约束1-24和1-25给定了卡车和无人机有效时间的初始值，约束1-26规定了起点仓库的访问次序一定在其他所有顾客节点之前，约束1-27和1-28给定了决策变量的取值范围，约束1-29规定了卡车破子圈辅助变量的取值范围，约束1-30和1-31规定了卡车和无人机的有效时间必须是非负实数，约束1-32给定了卡车访问次序辅助变量的取值范围。

在约束 equations (1-12) to (1-19) 中， $M \geq \max\{t_{c+1}, t'_{c+1}\}$ 取一个非常大的数，需要大于等于最后到达终点仓库的卡车（无人机）的有效时间。由于无法事先确定最小可接受的 M 值，因此一种方法是用 nearest neighbor heuristic 来计算一个访问所有顾客节点并返回仓库的时间上限。算法的大致过程：初始化 $M = 0$ ，然后从仓库开始构建卡车路径 ($i = 0$)，找到最近的还未访问过的顾客节点 j ，让 $M \leftarrow M + \tau_{ij}$ ，更新 $i = j$ 然后重复这个过程，即不断添加距离最近的未访问过的顾客节点直到所有的顾客都被访问一遍，最终让 $M \leftarrow M + \tau_{i,c+1}$ ，即让卡车返回仓库。

1.1.1 Flying Sidekick Traveling Salesman Problem with Multiple Drops

Gonzalez-R 等 (2020)^[5] 最早提出了类型为 Flying Sidekick Traveling Salesman Problem with Multiple Drops (FSTSP-MD) 的 MILP 数学模型，在文章中，作者称之为 truck-drone team logistic (TDTL)。该问题聚焦于 FSTSP 问题的延伸，即考虑无人机可以在单次起飞降落的过程（在 Murray 等 (2015)^[1] 中被称为 sortie）服务多个顾客节点的情况，但是不同于 FSTSP，TDTL 的起始节点和终止节点不是同一个节点。

关键假设如下：

- 不考虑卡车的最大航行里程，即卡车在整个运输过程中不需要补充燃料。
- 无人机和卡车只能在顾客节点会合，而不能够在任意的中间路程中进行会合。无人机和卡车在会合点通过更换电池使无人机充满电。无人机和卡车在会合点需要进行时间对齐，即如果无人机先到达会合点则需要等待卡车到达，反之亦然。
- 对于无人机更换电池的次数没有限制，无人机更换电池的服务时间相对于总体的时间而言可以忽略不计，因此不考虑更换电池的时间。由于无人机在每次和卡车会合时是通过更换电池来达到充电的目的，所以每次无人机在和卡车会合后起飞时，无人机的电池是满的。
- 只要无人机有充足的电量，无人机就会在服务过程中访问多个顾客直到下一次和卡车在顾客节点会合。
- 当无人机搭载在卡车上进行访问顾客节点时，无人机不消耗电量。
- 节点之间的距离用欧几里得距离来衡量，无人机和卡车都以恒定的速度行驶。
- 为了简化模型，假定无人机的电量消耗是线性的，即无人机的电量不会随着起飞降落或者携带的货物重量而进行非线性的变化。
- 每个顾客必须被无人机或者卡车服务一次。

TDTL 数学模型的符号含义如表1-2所示。

表 1-2: TDIL 模型符号及含义

| 符号 | 含义 |
|----------------|--|
| $G = \{N, A\}$ | 定义了要访问的节点集合和连接它们的有向链接集合的图。 |
| N | 图 G 的节点集合。 |
| $o \in N$ | 任务的起始节点。 |
| $e \in N$ | 任务的结束节点。 |
| A | 图 G 中的有向链接集合。 |
| $\delta^+(i)$ | 可以从节点 $i \in N$ 通过链接 A 到达的节点集合。 |
| $\delta^-(i)$ | 可以通过链接 A 到达节点 $i \in N$ 的节点集合。 |
| Q | 无人机电池的最大充电水平 (以时间单位表示), 即电池更换时的电量。 |
| t_{ij}^T | 链接 $(i, j) \in A$ 上卡车的行驶时间。 |
| t_{ij}^D | 链接 $(i, j) \in A$ 上无人机的行驶时间。 |
| M | 一个足够大的常数。 |
| u_{ij} | 二元变量, 如果链接 $(i, j) \in A$ 被卡车遍历, 则等于 1。 |
| v_{ij} | 二元变量, 如果链接 $(i, j) \in A$ 被无人机遍历, 则等于 1。 |
| s_i | 连续变量, 衡量节点 $i \in N$ 被卡车或无人机服务后的最早出发时间。 |
| b_i^- | 连续变量, 衡量无人机到达节点 $i \in N$ 时的电池电量。 |
| b_i^+ | 连续变量, 衡量无人机离开节点 $i \in N$ 时的电池电量。 |

TDIL 数学模型可以表示为 MILP 1.2。

Model 1.2: TDIL MILP

$$\min s_e \quad (1-33)$$

$$\text{s.t. } \sum_{j \in \delta^-(i)} u_{ji} \leq 1, \quad \forall i \in N \setminus \{o, e\} \quad (1-34)$$

$$\sum_{j \in \delta^+(i)} u_{ij} - \sum_{j \in \delta^-(i)} u_{ji} = 0, \quad \forall i \in N \setminus \{o, e\} \quad (1-35)$$

$$\sum_{j \in \delta^+(o)} u_{oj} = 1 \quad (1-36)$$

$$\sum_{i \in \delta^-(e)} u_{ie} = 1 \quad (1-37)$$

$$\sum_{j \in \delta^-(i)} v_{ji} \leq 1, \quad \forall i \in N \setminus \{o, e\} \quad (1-38)$$

$$\sum_{j \in \delta^+(i)} v_{ij} - \sum_{j \in \delta^-(i)} v_{ji} = 0, \quad \forall i \in N \setminus \{o, e\} \quad (1-39)$$

$$\sum_{j \in \delta^+(o)} v_{oj} = 1 \quad (1-40)$$

$$\sum_{i \in \delta^-(e)} v_{ie} = 1 \quad (1-41)$$

$$\sum_{i \in \delta^-(j)} u_{ij} + \sum_{i \in \delta^-(j)} v_{ij} \geq 1, \quad \forall j \in N \setminus \{o\} \quad (1-42)$$

$$s_j \geq s_i + t_{ij}^T \cdot u_{ij} - M \cdot (1 - u_{ij}), \quad \forall (i, j) \in A \quad (1-43)$$

$$s_j \geq s_i + t_{ij}^D \cdot v_{ij} - M \cdot (1 - v_{ij} + u_{ij}), \quad \forall (i, j) \in A \quad (1-44)$$

continued

$$s_o = 0 \quad (1-45)$$

$$b_j^- \leq Q + M \cdot (2 - v_{ij} - u_{ij}), \quad \forall (i, j) \in A \quad (1-46)$$

$$b_j^- \geq Q - M \cdot (2 - v_{ij} - u_{ij}), \quad \forall (i, j) \in A \quad (1-47)$$

$$b_j^+ \leq Q + M \cdot (2 - v_{ij} - u_{ij}), \quad \forall (i, j) \in A \quad (1-48)$$

$$b_j^+ \geq Q - M \cdot (2 - v_{ij} - u_{ij}), \quad \forall (i, j) \in A \quad (1-49)$$

$$b_j^- \leq b_i^+ - t_{ij}^D + M \cdot \left(1 - v_{ij} + u_{ij} + \sum_{k \neq i} u_{kj} \right), \quad \forall (i, j) \in A \quad (1-50)$$

$$b_j^- \geq b_i^+ - t_{ij}^D - M \cdot \left(1 - v_{ij} + u_{ij} + \sum_{k \neq i} u_{kj} \right), \quad \forall (i, j) \in A \quad (1-51)$$

$$b_j^+ \leq b_j^- + M \cdot \left(1 - v_{ij} + u_{ij} + \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-52)$$

$$b_j^+ \geq b_j^- - M \cdot \left(1 - v_{ij} + u_{ij} + \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-53)$$

$$b_j^- \leq b_j^+ - t_{ij}^D + M \cdot \left(1 - v_{ij} + u_{ij} + 1 - \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-54)$$

$$b_j^- \geq b_j^+ - t_{ij}^D - M \cdot \left(1 - v_{ij} + u_{ij} + 1 - \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-55)$$

$$b_j^+ \leq Q + M \cdot \left(1 - v_{ij} + u_{ij} + 1 - \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-56)$$

$$b_j^+ \geq Q - M \cdot \left(1 - v_{ij} + u_{ij} + 1 - \sum_{k \neq i} u_{kj} \right), \quad (i, j) \in A \quad (1-57)$$

$$b_0^+ = Q \quad (1-58)$$

目标函数1-33追求最小化服务完所有顾客并到达终止节点的时间 s_e 。

- 卡车相关的路径约束：约束 1-34 限制每个顾客节点最多只能被卡车服务一次；约束 1-35 要求卡车在顾客节点满足流入和流出相等的流平衡；约束 1-36 和 1-37 分别要求卡车只能在起始节点流出一次，只能在终止节点流入一次。
- 无人机相关的路径约束：约束 1-38 限制每个顾客节点最多只能被无人机服务一次；约束 1-39 要求无人机在顾客节点满足流入和流出相等的流平衡；约束 1-40 和 1-41 分别要求无人机只能在起始节点流出一次，只能在终止节点流入一次。
- 节点访问约束：约束 1-42 要求每个节点都必须被卡车或无人机（或者两者一起）访问过一次。
- 同步相关的约束：约束 1-43 和 1-44 计算了每个节点的离开时间；约束 1-45 规定了离开起始节点的时间。
- 无人机电量相关约束：约束 1-58 表示无人机离开起始节点 o 时的电量为 Q 。与无人机电量相关的约束根据无人机的状态不同可以分为三种类型，如图1-1所示。
 - a) 卡车搭载无人机：约束 equations (1-46) to (1-49) 表示当无人机和卡车同时访问节点 j 时，

即卡车搭载无人机进行服务，无人机在到达节点 j 和离开节点 j 时的状态都是充满电的状态 ($b_j^- = b_j^+ = Q$)，即卡车搭载无人机访问时无人机不消耗电量。

- b) 无人机单独服务节点：约束 equations (1-50) to (1-53) 表示当无人机单独访问节点 j 时，无人机在到达节点 j 的电量应该是无人机在离开节点 i 的电量减去路径 (i, j) 所消耗的电量 ($b_j^- = b_i^+ - T_{ij}^D$)，当无人机离开节点 j 时的电量应该等于无人机到达节点 j 的电量 ($b_j^+ = b_j^-$)，因为假设无人机在降落和起飞过程中不消耗电量。
- c) 无人机与卡车会合：约束 equations (1-54) to (1-57) 表示当无人机与卡车在节点 j 会合时，无人机到达节点 j 时的电量应该是 $b_j^- = b_i^+ - T_{ij}^D$ ，当无人机与卡车会合时，无人机会在卡车上通过替换电池来将电量充满，因此当无人机离开节点 j 时的电量应该是 $b_j^+ = Q$ 。

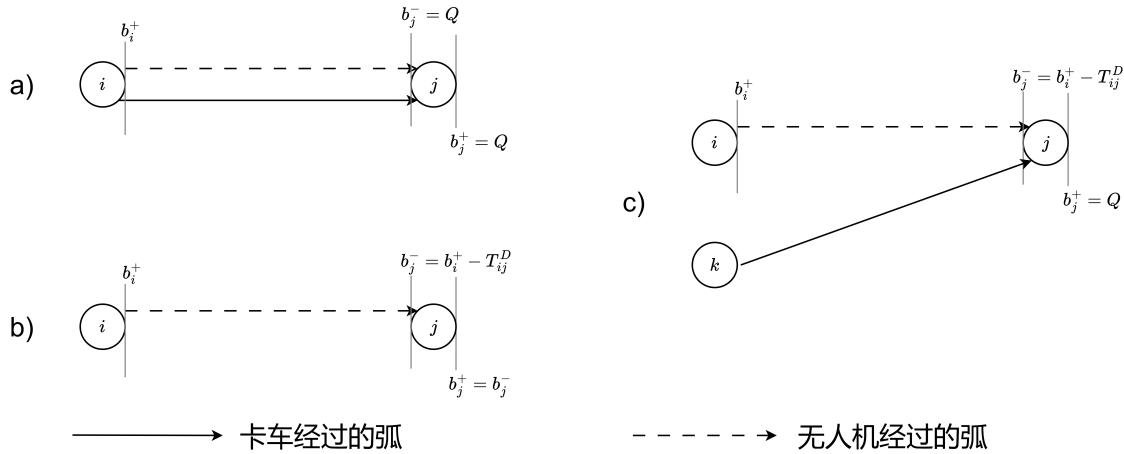


图 1-1: TDTL 模型无人机电量约束类型示意图

Liu 等 (2021)^[6] 同样提出了类型为 FSTSP-MD 的 MILP 数学模型。在文章中作者称之为 Two-Echelon Routing Problem for the Truck and Drone (2E-RP-T&D)。但是不同于 Gonzalez-R 等 (2020)^[5] 提出的 TDTL 模型，该问题考虑了不同重量的包裹对于无人机飞行消耗能源的影响。和 FSTSP 及 TDTL 一样，同样假设一辆卡车搭载一架无人机进行包裹递送，卡车的载重和续航没有进行限制，即卡车能够运送所有的包裹。该问题同样假设无人机和卡车在顾客点（或仓库）会合，但是该问题要求卡车必须先于无人机到达会合点（出于安全考虑）。2E-RP-T&D 的目标是最小化无人机和卡车的总成本。

Windras Mara 等 (2022)^[7] 在 Gonzalez-R 等 (2020)^[5] 的基础上提出了 multi-drop FSTS，但是不同于 Gonzalez-R 等 (2020)^[5] 的假设，该模型考虑了起飞和降落的服务时间。

multi-drop FSTSP 数学模型的符号含义如表 1-3 所示。

表 1-3: multi-drop FSTSP 数学模型符号及含义

| 符号 | 含义 |
|---|--|
| n | 顾客的数量 |
| k | 操作的索引 |
| $G = (V, A)$ | 定义了要访问的节点集合 V 和连接它们的弧集合 A 的图 |
| V | 图 G 的节点集合, 包括一个仓库节点 $\{0\}$ 和 n 个客户节点 $N = \{1, 2, \dots, n\}$ |
| A | 图 G 中的弧集合, 连接节点对 (i, j) , 其中 $i, j \in V$ |
| $Vd \subseteq V$ | 可以由无人机服务的节点子集 |
| N | 客户节点集合, 即 $N = \{1, 2, \dots, n\}$ |
| $K = [OP_1, \dots, OP_k]$ | 所有可行操作的集合 |
| C | 卡车路径的非负节点数组 |
| D | 无人机路径的非负节点数组 |
| $S \subseteq V$ | 节点的子集 |
| $OP_k = \{\mathcal{T}_k, \mathcal{D}_k\}$ | 第 k 次递送的顺序, 也被称为 operation, 其中 \mathcal{T}_k 是卡车在 OP_k 时必须访问的节点序列, \mathcal{D}_k 是无人机在 OP_k 时必须访问的节点序列 |
| $K(v)$ | 包含了节点 v 的所有可行操作集合 |
| $K^-(v)$ | 包含了以节点 v 为起始点或发射点的所有可行操作集合 |
| $K^+(v)$ | 包含了以节点 v 为终止点或会合点的所有可行操作集合 |
| $K^{\mathcal{T}}(v)$ | 包含了以卡车访问节点 v 的所有可行操作集合 |
| $K^{\mathcal{D}}(v)$ | 包含了以无人机访问节点 v 的所有可行操作集合 |
| c_n | 顾客的需求 |
| $d_{i \rightarrow j}^{\mathcal{T}}$ | 卡车从节点 i 行驶到 j 的距离 |
| $d_{i \rightarrow j}^{\mathcal{D}}$ | 无人机从节点 i 行驶到 j 的距离 |
| $\tau_{i \rightarrow j}^{\mathcal{T}}$ | 弧 $(i, j) \in A$ 上卡车的行驶时间 |
| $\tau_{i \rightarrow j}^{\mathcal{D}}$ | 弧 $(i, j) \in A$ 上无人机的行驶时间 |
| s^L | 无人机的启动时间 |
| s^R | 无人机的回收时间 |
| ϵ | 无人机的续航时间 |
| $w_r^{\mathcal{T}}$ | 卡车在节点 j 的等待时间 |
| $w_r^{\mathcal{D}}$ | 无人机在节点 j 的等待时间 |
| $t(\mathcal{T}_k)$ | 卡车在 OP_k 中完成其对应任务所需的时间, 计算公式为 1-60 |
| $t(\mathcal{D}_k)$ | 无人机在 OP_k 中完成其对应任务所需的时间, 计算公式为 1-61 |
| $t_k = t(OP_k)$ | OP_k 的完成时间 (OP_k 的成本函数), 计算公式为 1-59 |
| x_k | 二元决策变量, 如果选择操作 k , 则等于 1, 否则等于 0 |
| Y_v | 二元辅助决策变量, 如果节点 v 被选为至少一个操作的起始节点, 则等于 1, 否则等于 0 |

Definition 1.1: time for operations

$$t(OP_k) = \begin{cases} \max\{t(\mathcal{T}_k), t(\mathcal{D}_k)\}, & \text{当 } OP_k \text{ 包含至少一个无人机访问的节点时} \\ t(\mathcal{T}_k), & \text{其他} \end{cases} \quad (1-59)$$

$$t(\mathcal{T}_k) = \begin{cases} s^L + \tau_{i \rightarrow C_1}^{\mathcal{T}} + \sum_{i=1}^{|C|-1} \tau_{C_i \rightarrow C_{i+1}}^{\mathcal{T}} + \tau_{C_{|C|} \rightarrow j}^{\mathcal{T}} + s^R, & \text{当 } D \neq \emptyset \text{ 且 } |C| > 1 \\ s^L + \tau_{i \rightarrow C_1}^{\mathcal{T}} + \tau_{C_1 \rightarrow j}^{\mathcal{T}} + s^R, & \text{当 } D \neq \emptyset \text{ 且 } |C| = 1 \\ \tau_{i \rightarrow j}^{\mathcal{T}}, & \text{其他} \end{cases} \quad (1-60)$$

$$t(\mathcal{D}_k) = \begin{cases} s^L + \tau_{i \rightarrow D_1}^{\mathcal{D}} + \sum_{i=1}^{|D|-1} \tau_{D_i \rightarrow D_{i+1}}^{\mathcal{D}} + \tau_{D_{|D|} \rightarrow j}^{\mathcal{D}} + s^R, & \text{当 } |D| > 1 \\ s^L + \tau_{i \rightarrow D_1}^{\mathcal{D}} + \tau_{D_1 \rightarrow j}^{\mathcal{D}} + s^R, & \text{其他} \end{cases} \quad (1-61)$$

multi-drop FSTSP 数学模型可以表示为 MILP 1.3。

Model 1.3: multi-drop FSTSP MILP

$$\min M = \sum_{k \in K} t_k x_k \quad (1-62)$$

$$\text{s.t. } \sum_{k \in K(v)} x_k \geq 1, \quad \forall v \in V \setminus \{0\} \quad (1-63)$$

$$\sum_{k \in K^{\mathcal{D}}(v)} x_k \leq 1, \quad \forall v \in Vd \setminus \{0\} \quad (1-64)$$

$$\sum_{k \in K^{\mathcal{T}}(v)} x_k \leq 2 \left(1 - \sum_{k \in K^{\mathcal{D}}(v)} x_k \right), \quad \forall v \in Vd \setminus \{0\} \quad (1-65)$$

$$\sum_{k \in K^+(v)} x_k \leq nY_v, \quad \forall v \in V \quad (1-66)$$

$$\sum_{k \in K^+(v)} x_k = \sum_{k \in K^-(v)} x_k, \quad \forall v \in V \quad (1-67)$$

$$\sum_{k \in K^+(S)} x_k \geq Y_v, \quad \forall S \subset V \setminus \{0\}, v \in S \quad (1-68)$$

$$\sum_{k \in K^+(0)} x_k \geq 1 \quad (1-69)$$

$$\tau_{i \rightarrow D_1}^{\mathcal{D}} + \sum_{i=1}^{|D|-1} \tau_{D_i \rightarrow D_{i+1}}^{\mathcal{D}} + \tau_{D_{|D|} \rightarrow j}^{\mathcal{D}} + w_r^{\mathcal{D}} \leq \epsilon \quad (1-70)$$

$$Y_0 = 1 \quad (1-71)$$

$$x_k \in \{0, 1\}, \quad \forall k \in K \quad (1-72)$$

$$Y_v \in \{0, 1\}, \quad \forall v \in V \quad (1-73)$$

目标函数 1-62 旨在最小化总路径的完成时间。

- 顾客相关的约束:** 约束 1-63 确保所有的节点都被至少访问过一次；约束 1-64 确保所有可以被无人机服务的顾客点最多被无人机服务一次；约束 1-65 确保被无人机服务过的节点不会被卡车服务，反之亦然，系数 2 是由于当无人机没有服务过节点 v 时，节点 v 可以作为卡车某个操作的终点及另一个操作的起点（即在节点 v 最多在两个操作中存在，一次作为操作的起点，一次作为操作的终点）；约束 1-66 表示如果节点 v 是某个操作的终止节点，那么节点 v 一定要是某

个操作的起始节点。

- Hamilton 图^[8]约束： 约束 1-67 确保以节点 v 为终止点的操作接下来的操作会以节点 v 为起始点（流量平衡约束）；约束 1-68 确保任何非空的客户子集 S 必须至少有一个操作从外部进入，即如果节点 v 被选为某个操作的起始节点 ($Y_v = 1$)，则必须存在至少一个操作进入包含 v 的子集 S ，避免子回路的产生（消除子回路约束）；约束 1-69 确保终止点为仓库的操作只有一次，这是因为如果终止点为仓库的操作大于 1 次，由于约束 1-65 限制了卡车和无人机服务节点的次数，并且约束 1-63 确保每个节点都被访问过至少一次，而目标函数是最小化时间成本，如果多次以仓库为终止点的操作存在，则会增加成本，因此终止节点为仓库的操作大于 1 次的情况会被过滤掉。
- 无人机相关的约束： 约束 1-70 要求无人机在一次 sortie 过程中消耗的电量不能超过其续航限制。
- 决策变量和辅助决策变量及初始值： 约束 1-71 通过确保仓库被作为起点访问使得仓库作为路径的起始点；约束 1-72 和 1-73 定义了决策变量 x_k 和辅助决策变量 Y_v 的取值范围。

Windras Mara 等 (2022)^[7] 用 Adaptive Large Neighborhood Search (ALNS) 来解决该问题，文章中使用到的 ALNS 算法符号及含义如 1-4 所示。

Definition 1.2: acceptance criterion

$$\Pr = \begin{cases} \exp\left(\frac{-(M' - M)}{BZxT_e}\right), & \text{for } M' - M > 0 \\ 1, & \text{for } M' - M \leq 0 \end{cases} \quad (1-74)$$

$$\Pr(c^-) = \frac{\omega_c^-}{\sum_{q \in Q} \omega_q^-}, \quad \forall c \in Q \quad (1-75)$$

$$\Pr(c^+) = \frac{\omega_c^+}{\sum_{r \in R} \omega_r^+}, \quad \forall c \in R \quad (1-76)$$

$$\omega_{i+1}^- = \lambda(\omega_i^-) + (1 - \lambda)(\eta_i) \quad (1-77)$$

$$\eta_i = \begin{cases} \Omega_1, & \text{if } M_i \leq M^* \\ \Omega_2, & \text{if } M_i \leq M_{i-1} \\ \Omega_3, & \text{if } M_i > M_{i-1} \wedge U(0, 1) < \Pr \\ \Omega_4, & \text{if } M_i > M_{i-1} \end{cases} \quad (1-78)$$

$$f_c = f_1 - (f_1 - f_0) \left(\frac{i_c}{I} \right), \quad \forall c \in \{1, \dots, I\} \quad (1-79)$$

$$ND_c = n(f_c), \quad \forall c \in \{1, \dots, I\} \quad (1-80)$$

$$(1-81)$$

1.2 Parallel Drone Scheduling Traveling Salesman Problem

Parallel Drone Scheduling Traveling Salesman Problem (PDSTSP) 同样由 Murray 等 (2015)^[1] 提出。PDSTSP 适用于大量的顾客节点在无人机直接从仓库起飞的续航里程范围内。PDSTSP 描述：一辆卡车和一群（单个或者多个都可以）完全相同的无人机从单个仓库出发分别服务顾客，每个顾客只能被服务一次，卡车遵循 TSP 路径服务，无人机直接从仓库起飞服务顾客，不同于 FSTSP，PDSTSP 中的无人机不需

表 1-4: ALNS 算法使用到的符号及含义

| 符号 | 含义 |
|--------------------------------|---|
| s^* | 最佳可行解 |
| s' | 新的可行解 |
| M | 目标函数值 |
| M' | 新的目标函数值 |
| M^* | 最优目标函数值 |
| Q | destroy operators 集合 |
| R | repair operators 集合 |
| ω^- | destroy operator 权重 |
| ω^+ | repair operator 权重 |
| η_I | destroy/repair operators 在当前迭代 I 中的表现 |
| Pr | 选择的概率, 计算公式为 1-74 |
| T_c | 当前的温度 |
| T_1 | 初始的温度 |
| T_0 | 最终的温度 |
| α | 冷却系数 |
| BZ | Boltzmann 常数 |
| $\Omega_1, \Omega_2, \Omega_3$ | 权重调整参数 |
| λ | 衰减参数 |
| f_1 | 初始的 destroy ratio |
| f_0 | 最终的 destroy ratio |
| I_T | 在某一温度下的迭代次数 |
| P | 概率 |
| f_c | 递减线性函数 |
| ND_c | 被 destroyed 的顾客节点数量, 由 1-79 和 1-80 决定 |
| AR_s | 便于归档 repair 的归档解 |
| RD | Destroy operator - Random removal |
| WTD | Destroy operator - Worst truck distance removal |
| WDD | Destroy operator - Worst drone distance removal |
| BD | Destroy operator - Block destroy removal |
| RR | Repair operator - Random repair |
| AR | Repair operator - Archive repair |
| LDR-1 | Repair operator - Least distance repair $d_{i-1 \rightarrow i}$ |
| LDR-2 | Repair operator - Least distance repair $d_{i \rightarrow i+1}$ |
| HR | Repair operator - Hill climbing |

要和卡车进行会合。PDSTSP 的目标是最小化最终到达仓库的卡车（无人机）的时间。

PDSTSP 数学模型的符号含义如表1-5，基本上沿用了 FSTSP 的符号。

表 1-5: PDSTSP 模型符号及含义

| 符号 | 含义 |
|---|---------------------------------------|
| 0 | 起点仓库 |
| $c + 1$ | 终点仓库 |
| $C = \{1, 2, \dots, c\}$ | 全部客户集合 |
| $C' \subseteq C$ | 可以接受无人机访问的客户集合 ^a |
| $C'' \subseteq C'$ | 在无人机的航行范围内可以接受无人机服务的顾客集合 ^b |
| $N_0 = \{0, 1, 2, \dots, c\}$ | 流出节点集合 |
| $N_+ = \{1, 2, \dots, c + 1\}$ | 流入节点集合 |
| $N = \{0, 1, 2, \dots, c, c + 1\}$ | 全部节点集合 |
| $v \in V$ | 无人机集合 |
| $\tau'_{i,j} / \tau_{i,j}$ | 弧 $\langle i, j \rangle$ 的飞行/行驶时间成本 |
| $1 \leq \hat{u}_i \leq c + 2$ | 卡车破子圈辅助变量 |
| $\hat{y}_{i,v} \in \{0, 1\}, i \in C'', v \in V$ | 无人机访问决策变量 |
| $\hat{x}_{i,j} \in \{0, 1\}, i \in N_0, j \in \{N_+ : j \neq i\}$ | 卡车路由决策变量 |

^a 指包裹重量没有超过无人机的载重限制，不需要顾客签收，顾客的位置允许无人机起降等限制。

^b 当 $\tau'_{0,i} + \tau'_{i,c+1} \leq e$ 时，顾客 $i \in C'$ 属于集合 C'' 。

PDSTSP 数学模型可以表示为 MILP 1.4。

Model 1.4: PDSTSP MILP

$$\min z \quad (1-82)$$

$$\text{s.t. } z \geq \sum_{i \in N_0} \sum_{\substack{j \in N_+ \\ j \neq i}} \tau'_{i,j} \hat{x}_{i,j} \quad (1-83)$$

$$z \geq \sum_{i \in C''} (\tau'_{0,i} + \tau'_{i,c+1}) \hat{y}_{i,v}, \quad \forall v \in V \quad (1-84)$$

$$\sum_{\substack{i \in N_0 \\ i \neq j}} \hat{x}_{i,j} + \sum_{\substack{v \in V \\ j \in C''}} \hat{y}_{j,v} = 1, \quad \forall j \in C \quad (1-85)$$

$$\sum_{j \in N_+} \hat{x}_{0,j} = 1 \quad (1-86)$$

$$\sum_{i \in N_0} \hat{x}_{i,c+1} = 1 \quad (1-87)$$

$$\sum_{\substack{i \in N_0 \\ i \neq j}} \hat{x}_{i,j} = \sum_{\substack{k \in N_+ \\ k \neq j}} \hat{x}_{j,k}, \quad \forall j \in C \quad (1-88)$$

$$\hat{u}_i - \hat{u}_j + 1 \leq (c + 2)(1 - \hat{x}_{i,j}), \quad \forall i \in C, j \in \{N_+ : j \neq i\} \quad (1-89)$$

$$1 \leq \hat{u}_i \leq c + 2, \quad \forall i \in N_+ \quad (1-90)$$

$$\hat{x}_{i,j} \in \{0, 1\}, \quad \forall i \in N_0, j \in \{N_+ : j \neq i\} \quad (1-91)$$

$$\hat{y}_{i,v} \in \{0, 1\}, \quad \forall i \in C'', v \in V \quad (1-92)$$

目标函数1-82追求最小化完工时间 z , 即无人机和卡车最晚到达终点仓库的时间, 通过约束1-83和1-84分别限制卡车和无人机最晚到达终点仓库的时间来实现; 约束1-85确保了每个顾客能且只能被服务一次, 服务可以由无人机或者卡车提供; 约束1-86和1-87要求卡车必须从起点仓库0出发并返回终点仓库 $c+1$ 一次, 约束1-88要求卡车在中间的顾客节点满足流入和流出相等的流约束; 约束1-89是 MTZ 形式的破子圈约束; 约束1-90, 1-91和1-92给出了决策变量和辅助决策变量的取值范围。

Ham(2018)^[9] 在 PDSTSP 的基础上提出了 multi-truck, multi-drone, and multi-depot scheduling problem constrained by time-window, drop-pickup synchronization, and multi-visit, with the objective to minimize the maximum completion time over all tasks (PDSTSP^{DP}), 不同于 PDSTSP, 该问题考虑了无人机递送和收回货物的情况, 即当无人机递送货物到顾客返回时, 可以飞行到另一个需要退货的顾客节点进行回收货物以使无人机满载, 从而满足退货的需求。另外, 该问题还考虑了服务顾客的时间窗问题, 即顾客需求有时间限制, 并且可以在不同的时间段为同一个顾客服务。

References

- [1] MURRAY C C, CHU A G. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery[J/OL]. *Transportation Research Part C: Emerging Technologies*, 2015, 54: 86-109. DOI: [10.1016/j.trc.2015.03.005](https://doi.org/10.1016/j.trc.2015.03.005).
- [2] 运筹 OR 帷幄. 交通 | 带飞行助手的旅行商问题：无人机协助的配送优化建模及求解（附代码）[EB/OL]. 2024[2025-02-21]. <https://zhuanlan.zhihu.com/p/3235861366>.
- [3] 运筹 OR 帷幄. 优化 | 浅谈旅行商问题（TSP）的七种整数规划模型[EB/OL]. 2022 年 01 月 19 日 20:37[2025-02-23]. <https://mp.weixin.qq.com/s/tDYOxlSQHKRJkf5EcaBJ1A>.
- [4] 运筹 OR 帷幄. 优化 | TSP 中两种不同消除子环路的方法及 callback 实现（Python 调用 Gurobi 求解）[EB/OL]. 2020 年 09 月 26 日 20:30[2025-02-23]. https://mp.weixin.qq.com/s/i7I-o0LiC_JP3vVOQw2AIw.
- [5] GONZALEZ-R P L, CANCA D, ANDRADE-PINEDA J L, et al. Truck-drone team logistics: A heuristic approach to multi-drop route planning[J/OL]. *Transportation Research Part C: Emerging Technologies*, 2020, 114: 657-680. DOI: [10.1016/j.trc.2020.02.030](https://doi.org/10.1016/j.trc.2020.02.030).
- [6] LIU Y, LIU Z, SHI J, et al. Two-echelon routing problem for parcel delivery by cooperated truck and drone[J/OL]. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021, 51(12): 7450-7465. DOI: [10.1109/TSMC.2020.2968839](https://doi.org/10.1109/TSMC.2020.2968839).
- [7] WINDRAS MARA S T, RIFAI A P, SOPHA B M. An adaptive large neighborhood search heuristic for the flying sidekick traveling salesman problem with multiple drops[J/OL]. *Expert Systems with Applications*, 2022, 205: 117647. DOI: [10.1016/j.eswa.2022.117647](https://doi.org/10.1016/j.eswa.2022.117647).
- [8] WIKIPEDIA. Hamiltonian path - wikiwand[EB/OL]. [2025-03-20]. <https://www.wikiwand.com/en/articles/Hamiltonian%20path>.
- [9] HAM A M. Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming[J/OL]. *Transportation Research Part C: Emerging Technologies*, 2018, 91: 1-14. DOI: [10.1016/j.trc.2018.03.025](https://doi.org/10.1016/j.trc.2018.03.025).