An Exact Algorithm for Heterogeneous Drone-Truck Routing Problem

Munjeong Kang, Chungmok Lee*
*Email : chungmok@hufs.ac.kr

Department of Industrial and Management Engineering
Hankuk University of Foreign Studies

2020 석사논문경진대회

The potential use of drones in logistics

Attempts to use drones for parcel delivery service

Amazon's Prime Air
(source: amazon.com)

DHL's Parcelcopter
(source: dhl.com)

Workhouse's HorseFly
(source: workhouse.com)

The Benefits and Shortcomings of a drone-based delivery system

Benefits

- Have a low per-mile cost
- Operate without human intervention
- Travel at high speeds while being unaffected by road traffic

Shortcomings

- Have an extremely low carrying capacity and short travelling radius
- Necessitate frequent returns to a central depot.

The drone-truck cooperation routing

Operation characteristics of drone and truck

Drone	**Weight ↓**	**Speed ↑**	**Capacity ↓**	**Range ↓**	**Reliability ↓**
Truck	**Weight ↑**	**Speed ↓**	**Capacity ↑**	**Range ↑**	**Reliability ↑**

Two distinct approaches in the literature

(1) The variants of the FSTSP, and (2) The drone station-based approaches

- **The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery, Murray, C. C. and Chu, A. G. (2015)**

  The authors presented a mathematical formulation and proposed a route-construction type heuristic.

- **Drone delivery from trucks: Drone scheduling for given truck routes, Boysen, N., Briskorn, D., Fedtke, S. and Schwerdfeger, S. (2018)**

  Boysen et al. considered the drone scheduling problem (DSP), which determines the drone route from a given truck route.

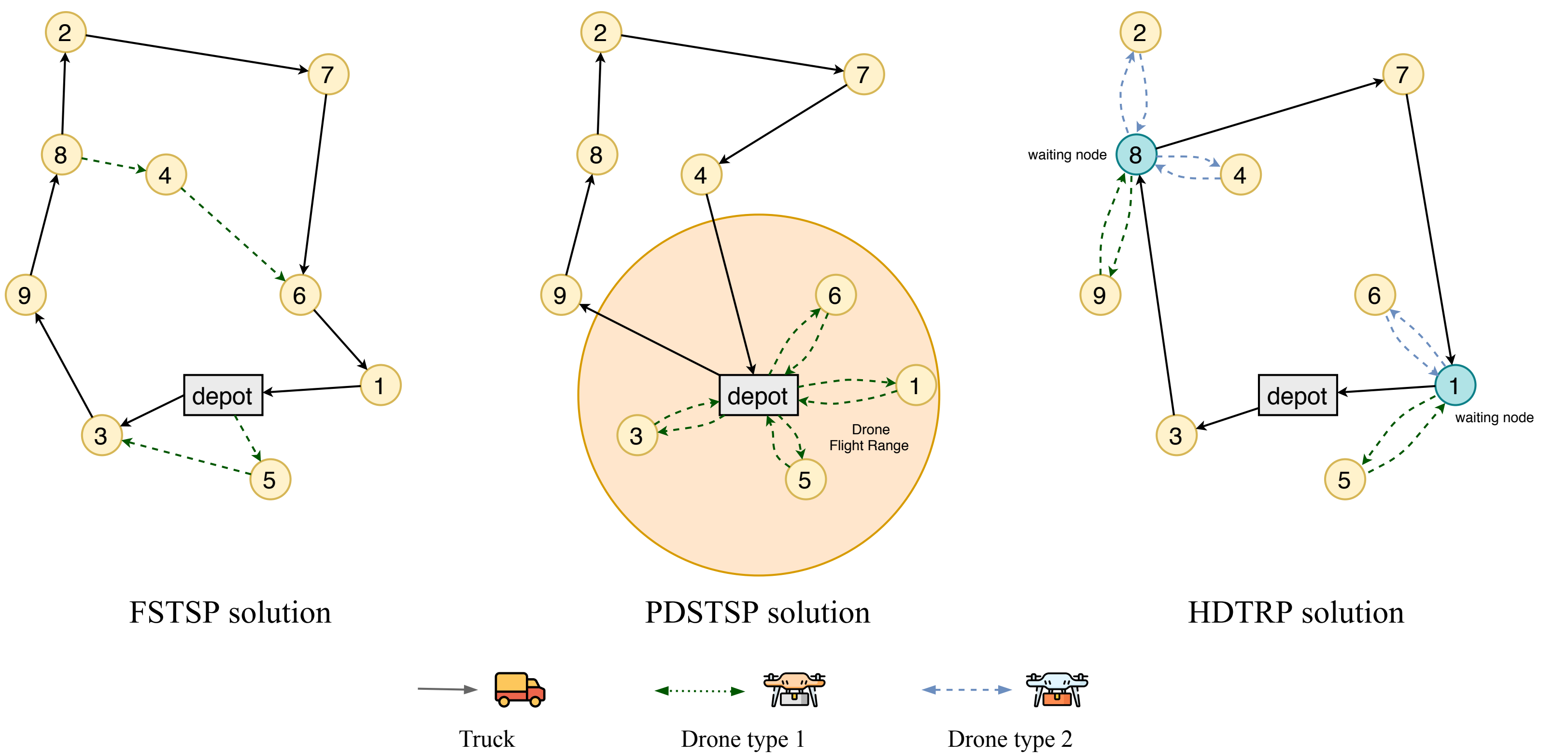
- **Traveling salesman problem with a drone station, Kim, S. and Moon, I. (2019)**

  With a given set of drone stations, the authors developed an optimization algorithm by deriving a decomposition method.

- **Matheuristic algorithms for the parallel drone scheduling traveling salesman problem, DellAmico, M., Montemanni, R. and Novellani, S. (2020)**

  Two heuristic algorithms for the PDSTSP were developed by DellAmico et al.

Heterogeneous Drone-Truck Routing Problem (HDTRP)

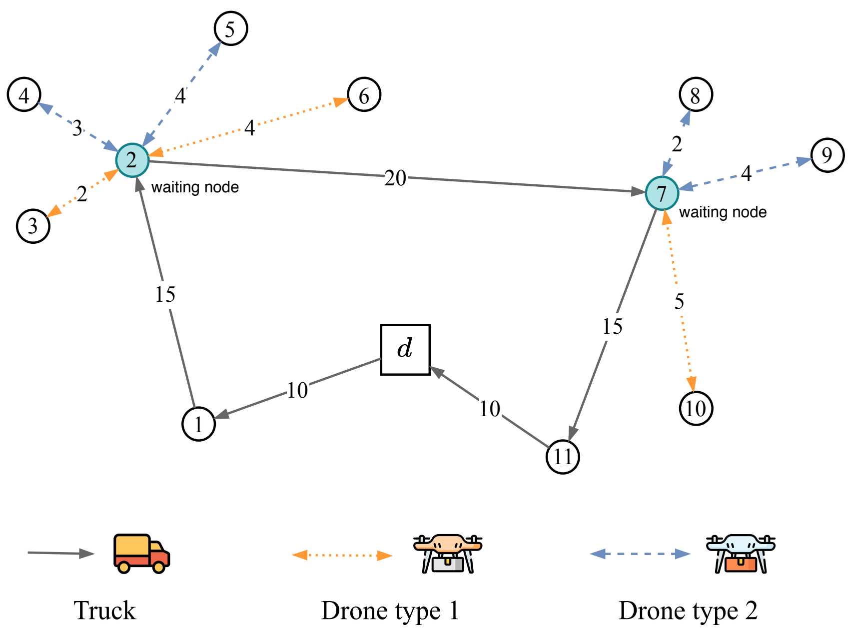


- **The HDTRP addresses the drawbacks of the previous approaches by replacing the concept of drone stations with truck's temporary waiting.**
- **We consider heterogeneous drones that have different characteristics such as battery capacity and flight speed.**
- **We develop an exact algorithm based on the logic-based Benders decomposition approach.**

Example of the HDTRP

Assumptions

1. Each drone can carry a single demand.
2. Each drone has specific speed and battery capacity.
3. The truck has a sufficient capacity to deliver all demands while carrying all drones
4. Multiple drones can be dispatched for deliveries at the same time, and the drones must return to the location the drones depart from.
5. The truck can leave the node only after all drones return to the truck.



The goal is to minimize the total sum of truck travel and waiting times.

The total time for complete delivery is 83 (=70+7+6).

HDTRP Notation

Parameters

s/t : Duplicated depot nodes
N : Set of customers
N_s : $N \cup \{s\}$
N_t : $N \cup \{t\}$
A : Set of arcs $\{(i,j) \mid i \in N_s, j \in N_t, i \neq j\}$
B^l : Battery capacity of drone l
L : Number of drones
t_{ij}^v : Travel time of vehicle from i to j
t_{ij}^l : Travel time of drone l from i to j
s_j^v : Service time of vehicle node j
s_j^l : Service time of drone l node j
b_{ij} : Required battery when the drone delivers the demand for j from i

Decision variables

x_{ij} : 1 if the vehicle travels from i to j, 0 otherwise
y_{ij} : 1 if j is delivered from i by the drone, 0 otherwise
h_{ij}^l : 1 if j is delivered from i by the drone l, 0 otherwise
v_i : Visiting order of node i
w_i : Waiting time of node i

Mathematical Formulation : Problem (P)

$$\text{(P)} \quad \min \sum_{i \in N_s} w_i + \sum_{(i,j) \in A} (t_{ij}^v + s_j^v)\, x_{ij} \tag{1}$$

$$\text{s.t.} \quad \sum_{j \in N} x_{sj} = 1\,, \tag{2}$$

$$\sum_{i \in N} x_{it} = 1\,, \tag{3}$$

$$\sum_{j \in N_t : j \neq i} x_{ij} = \sum_{j \in N_s : j \neq i} x_{ji}\,, \qquad \forall i \in N, \tag{4}$$

$$v_i - v_j \leq M\big(1 - x_{ij}\big) - 1, \qquad \forall (i,j) \in A, \tag{5}$$

$$\sum_{i \in N_s : i \neq j} x_{ij} + \sum_{i \in N_s : i \neq j} y_{ij} \geq 1\,, \qquad \forall j \in N, \tag{6}$$

$$M \sum_{j \in N_t : j \neq i} x_{ij} \geq \sum_{j \in N : j \neq i} y_{ij}\,, \qquad \forall i \in N_s, \tag{7}$$

$$\sum_{i \in N_s} \sum_{j \in N : j \neq i} b_{ij}^l h_{ij}^l \leq B^l\,, \qquad \forall l \in L, \tag{8}$$

$$y_{ij} = \sum_{l \in L} h_{ij}^l\,, \qquad \forall (i,j) \in A, \tag{9}$$

$$w_i \geq \sum_{j \in N : j \neq i} \big(2t_{ij}^l + s_j^l\big) h_{ij}^l\,, \qquad \forall i \in N_s\; l \in L, \tag{10}$$

$$v_s = 0, \tag{11}$$

$$x_{ij} \in \{0, 1\}, \qquad \forall (i,j) \in A, \tag{12}$$

$$y_{ij} \in \{0, 1\}, \qquad \forall (i,j) \in A, \tag{13}$$

$$h_{ij}^l \in \{0, 1\}, \qquad \forall (i,j) \in A, l \in L. \tag{14}$$

Minimize the sum of the total waiting times, the travel times, and the service times. (1)

The well-known flow balance constraints. (2,3,4)

The sub-tour elimination constraints. (5)

At least one truck or drones must serve all nodes. (6)

The drones can be dispatched from the node i only if the truck visit the node. (7)

The total battery consumption cannot exceed the given battery capacity. (8)

Only one drone deliver to node j from node i. (9)

The waiting time should be greater than or equal to the total time spent by the drone deliveries. (10)

The problem (P) has many binary decision variables that make solving the formulation by the MIP solvers very challenging.

Logic-based Benders Decomposition Approach

The problem (P) consists of two distinct decisions: the truck route (i.e., x_{ij}) and drone deliveries (i.e., h_{ij}^{l}).
Without the coupling constraints (6) and (7), the problem can be separated into two independent decision problems.

The Benders Master Problem (BMP)

- Find the best discrete decision, which in turn is provided to the BSP.

The Benders Subproblem (BSP)

- The BSP is solved after "fixing" the discrete decision variables to assert the validity of the provided solution of the BMP.

➡ ***Optimality*** **cuts,** ***Feasibility*** **cuts**

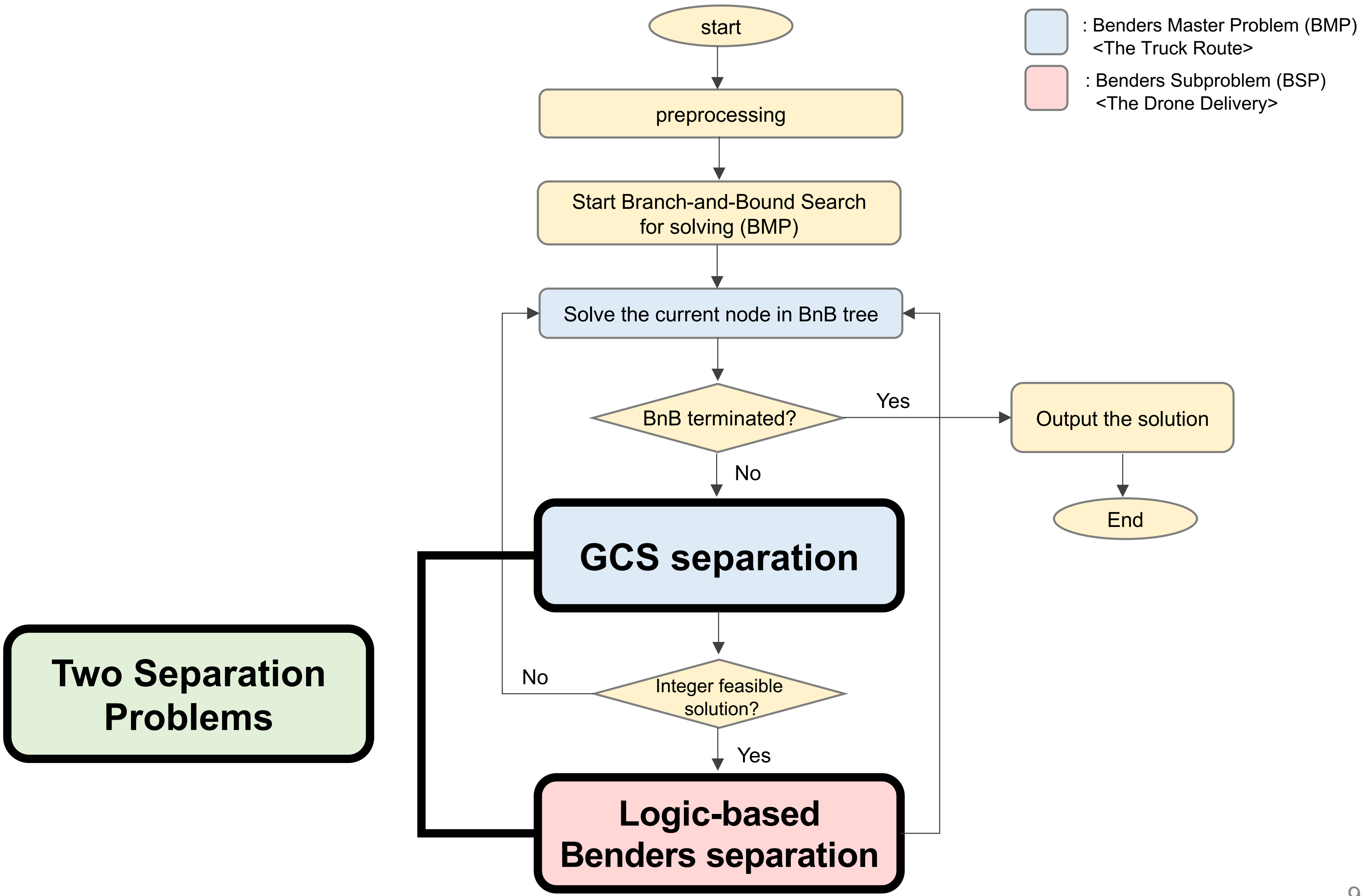
The Classical Benders Decomposition

- BSP should be a convex optimization problem because the Benders cuts are obtained from the duality of the BSP.
- However, we cannot employ the classical Benders decomposition approach for solving the problem (P). (∵ The BSP is an MIP problem)

The Logic-based Benders Decomposition

The main idea is to utilize "inference dual" that provides a valid lower bound of the objective value for the Benders master solutions.

HDTRP Flowchart : Branch-and-Cut Algorithm



Benders Master Problem (BMP)

The purpose of the problem (BMP) is to find a feasible truck route.

$$\text{(BMP)} \quad \min \sum_{(i,j)\in A} (t_{ij}^v + s_j^v)\, x_{ij} + W \tag{15}$$

$$\text{s.t. } (2)-(4),$$

$$\sum_{j\in N_t: j\neq i} x_{ij} = z_i\,, \qquad \forall i \in N, \tag{16}$$

$$\sum_{i\in N_s: i\neq j} \sum_{l\in L} h_{ij}^l \leq 1 - z_j\,, \qquad \forall j \in N, \tag{17}$$

$$z_i \geq h_{ij}^l, \qquad \forall (i,j) \in A, l \in L, \tag{18}$$

$$\sum_{i\in N_s, j\in N, j\neq i} b_{ij}^l h_{ij}^l \leq B^l\,, \qquad \forall l \in L, \tag{19}$$

$$W \geq \sum_{i\in N_s} w_i\,, \tag{20}$$

$$w_i \geq \sum_{j\in N: j\neq i} \left(2t_{ij}^l + s_j^l\right) h_{ij}^l\,, \forall i \in N_s\; l \in L, \tag{21}$$

$$x_{ij} \in \{0,1\}, \qquad \forall (i,j) \in A, \tag{22}$$

$$z_i \in \{0,1\}, \qquad \forall i \in N_s, \tag{23}$$

$$h_{ij}^l \geq 0, \qquad \forall (i,j) \in A, l \in L. \tag{24}$$

There is no sub-tour elimination constraints

The binary decision variable z_i is 1 if the truck visit node i, 0 otherwise.

Proposition 1. *The constraints (25) exclude no integer optimal h_{ij}^l solutions while strengthening the linear relaxation bound.*

$$w_i \geq \min_{l\in L}\{2t_{ij}^l + s_j^l\} \sum_{l\in L} h_{ij}^l\,, \forall (i,j) \in A. \tag{25}$$

GCS (Generalized Cutset Inequalities)

$$\sum_{(i,j)\in\delta^+(S)} x_{ij} \geq \sum_{(i,j)\in\delta^+(\{k\})} x_{ij}\,, \forall k \in S, S \subseteq N, |S| \geq 2, \tag{26}$$

where $\delta^+(S) \coloneqq \{(i,j) \in A \mid i \in S, j \notin S\}$

i.e., a set of arcs leaving the set S

Since there are exponentially many constraints (26), we use the branch-and-cut method.

GCS Separation

At every node during the branch-and-bound search for solving the problem (BMP), we solve the separation problem presented in Algorithm 1, which is adopted from Taccari (2016).

Algorithm 1 Separation of GCS for sub-tour elimination

$x^* \leftarrow$ solution of the (BMP) at the current BnB node
$\epsilon \leftarrow 0.8$
Construct graph $G(N_{st}, A^*)$, where $A^* := \{(i,j) \in A \mid x^*_{ij} > 0$ or $x^*_{ji} > 0\}$
$\mathcal{S} \leftarrow \{S \subseteq N_s \mid S$ is a strongly connected component on $G\}$ ▷ Depth-first search on $G(N_{st}, A^*)$
$C \leftarrow \emptyset$
for $S \in \mathcal{S}$ **do**
    for $k \in S$ **do**
        $v \leftarrow \sum_{(i,j)\in\delta^+(\{k\})} x^*_{ij} - \sum_{(i,j)\in\delta^+(S)} x^*_{ij}$
        if $v \geq \epsilon$ **then**
            $C \leftarrow C \cup \{(v, S, k)\}$
        end if
    end for
end for
return C

Benders Subproblem (BSP)

The goal of the BSP is to identify Benders cuts with a given Benders master solution.

Let (x^*, z^*, W^*) be a feasible truck solution of the (BMP), $N_0^*(z^*) := \{i \in N \mid z_i^* = 0\}$, $N_1^*(z^*) := \{i \in N \mid z_i^* = 1\}$

$$
\begin{aligned}
\text{(BSP)} \quad \min \; & \sum_{i \in N_s} w_i & (27)\\
\text{s.t.} \; & \sum_{i \in N_s} \sum_{j \in N: j \neq i} b_{ij}^l h_{ij}^l \le B^l, \quad \forall l \in L, & (28)\\
& w_i \ge \sum_{j \in N: j \neq i} (2t_{ij}^l + s_j^l) h_{ij}^l, \quad \forall i \in N_s \; l \in L, & (29)\\
& \sum_{i \in N_s: i \neq j} \sum_{l \in L} h_{ij}^l \ge 1 - z_j^*, \quad \forall j \in N, & (30)\\
& \sum_{j \in N: j \neq i} \sum_{l \in L} h_{ij}^l \le |N| z_i^*, \quad \forall i \in N, & (31)\\
& h_{ij}^l \in \{0, 1\}, \quad \forall (i,j) \in A, l \in L, & (32)\\
& w_i \ge 0, \quad \forall i \in N_s. & (33)
\end{aligned}
$$

Minimize the total waiting times. (27)

All customers in N_0^* served by the drones. (30)

Prevent dispatch of the drones if the truck does not visit the node. (31)

- The problem (BSP) is the 0-1 Multiple Knapsack Problem with side constraints which is NP-hard.
- A significant number of binary decision variables are fixed to zero due to the constraints (31).
- With a given z^*, solving (BSP) can result in two cases : **Infeasibility** and **Optimal Solution**

Benders Cuts

$Case\ 1:\ (BSP)\ is\ infeasible \rightarrow$ **Add the Benders *feasibility* cut.**

This means that, with the current z^*, the drones cannot complete the deliveries due to the shortage of battery.
The Benders ***feasibility* cut** is defined as follows:

$$\sum_{i \in N_1(z^*)} (1 - z_i) + \sum_{i \in N_0(z^*)} z_i \geq 1. \qquad (34)$$

$Case\ 2:\ (BSP)\ has\ an\ optimal\ solution \rightarrow$ **Add the Benders *optimality* cut.**

$\widetilde{W}_{z^*}$: Objective value of the problem (BSP) with a given z^*

$\underline{W}^p$: A lower bound of waiting times, for any $p \in \{\underline{p}, \dots, \overline{p}\}$

$W^* = \widetilde{W}_{z^*} \rightarrow$ no Benders cut

$W^* < \widetilde{W}_{z^*} \rightarrow$ Benders $optimality$ cut

We assume that, in the preprocessing stage, we obtained a lower bound of waiting times $\underline{W}^p$ for all solutions in Z^p, i.e., for any$p \in \{\underline{p}, \dots, \overline{p}\}$, $\underline{W}^p$ satisfies :

$$\underline{W}^p \leq \widetilde{W}_z, \qquad \forall z \in Z^p \qquad (35)$$

The Benders ***optimality* cut** is defined as follows:

$$W \geq \widetilde{W}_{z^*} - \Omega(z^*) \sum_{i \in N_0(z^*)} z_i, \qquad (36)$$

$$where \quad \Omega(z^*) = \begin{cases} \max\limits_{q = p^*+1, \dots, \overline{p}} \left\{ \dfrac{\widetilde{W}_{z^*} - \underline{W}^q}{q - p^*} \right\}, if\ p^* < \overline{p} \\ \widetilde{W}_{z^*} - \underline{W}^{p^*}, otherwise \end{cases} \quad and \quad p^* = \textstyle\sum_{i \in N} z_i^*$$

Theorem

The constraint (36) is a valid Benders optimality cut, if the following condition is satisfied :

$$(C) \quad \underline{W}^{p} \le \underline{W}^{q}, \qquad \forall\, \underline{p} \le p < q \le \overline{p}$$

Proof. Let $\beta_{z^*}(z)$ denote the right-hand-side of the constraint (36), i.e.,

$$\beta_{z^*}(z) = \widetilde{W}_{z^*} - \Omega(z^*) \sum_{i \in N_0(z^*)} z_i,$$

(B1) $\widetilde{W}_z \ge \beta_{z^*}(z)$ for all feasible solution z of the Benders master problem (BMP)

(B2) $\widetilde{W}_z = \beta_{z^*}(z)$

The above conditions state that $\beta_{z^*}(z)$ should be a (tight) lower bound function of z. For the condition (B1), assume that there is $\hat{z}$, which is a feasible solution of (BMP), such that $\widetilde{W}_{\hat{z}} < \beta_{z^*}(\hat{z})$.

Preprocessing

The purpose of the preprocessing

- To accelerate the branch-and-bound search by providing a good incumbent solution
- To limit the length of the truck route so that the unnecessary search is avoided
- To provide the lower bound of the total waiting times $\underline{W}^p$ for the Benders optimality cuts.

$$\text{(LW-}p\text{)} \quad \min \sum_{i \in N_s} w_i \tag{37}$$

$$\text{s.t.} \sum_{i \in N_s} \sum_{j \in N : j \neq i} b_{ij}^l h_{ij}^l \leq B^l, \quad \forall l \in L, \tag{38}$$

$$w_i \geq \sum_{j \in N : j \neq i} (2t_{ij}^l + s_j^l) h_{ij}^l, \forall i \in N_s \; l \in L, \tag{39}$$

$$y_{ij} = \sum_{l \in L} h_{ij}^l, \quad \forall (i,j) \in A, \tag{40}$$

$$z_i \geq y_{ij}, \quad \forall i \in N_s, j \in N, i \neq j, \tag{41}$$

$$\sum_{i \in N} z_i = p, \tag{42}$$

$$\sum_{i \in N_s : i \neq j} y_{ij} + z_j \geq 1, \quad \forall j \in N, \tag{43}$$

$$z_s = 1, \tag{44}$$

$$z_i \in \{0,1\}, \quad \forall i \in N_s, \tag{45}$$

$$y_{ij} \in \{0,1\}, \quad \forall (i,j) \in A, \tag{46}$$

$$h_{ij}^l \in \{0,1\}, \quad \forall (i,j) \in A, l \in L. \tag{47}$$

Algorithm 2 Primal Heuristic

```
1: P ← {1, ..., |N| − 1}
2: for p ∈ P do
3:     Solve (LW-p)
4:     if (LW-p) is feasible then
5:         W^p ← objective value of (LW-p)
6:         z^p ← solution of (LW-p)
7:         Solve TSP with z^p. Let T^p be the objective value.
8:         Ẑ ← T^p + W^p                                  ▹ Primal heuristic solution
9:         p̲ ← p                                          ▹ Minimum truck route length
10:        Break
11:    end if
12: end for
13: return Ẑ, p̲
```

Algorithm 3 Truck Route Length Bounding

```
Require: Ẑ, p̲ from Algorithm 2
1: P ← {p̲, ..., |N| − 1}
2: for p ∈ P do
3:     Solve the root relaxation of (P-p)
4:     if (P-p) is feasible then
5:         Z̲^p ← objective value of the root relaxation of (P-p)
6:     end if
7: end for
8: p̲ ← min{p ∈ P | Z̲^p exists and Z̲^p < Ẑ}           ▹ Minimum truck route length
9: p̄ ← max{p ∈ P | Z̲^p exists and Z̲^p < Ẑ}           ▹ Maximum truck route length
10: return p̲, p̄
```

Experiments Environment

Experiments Setting

- Linux machine equipped with Intel i9-9900KS 5GHz CPU and 64GB RAM
- The algorithm was implemented by Python 3.7
- CPLEX 12.10 was used for solving the mathematical formulations

Test Instances

- VRP problems by Solomon(1987) and Augerat(1995)
- To address the trade-off between flight speed, service speed, and battery capacity, we introduce three parameters: α^l, β^l, B^l

$$t_{ij}^l \leftarrow \alpha^l t_{ij}^v, \qquad \forall (i,j) \in A, \tag{50}$$

$$s_i^l \leftarrow \beta^l s_i^v, \qquad \forall i \in N, \tag{51}$$

$$b_{ij}^l \leftarrow 2t_{ij}^l + s_j^l, \qquad \forall (i,j) \in A. \tag{52}$$

- For a drone $l \in L$, the smaller α^l, the faster the drone is. Similarly, the parameter β^l controls the drone's relative service speed
- b_{ij}^l represents the battery consumption for a round-trip, so the flight time is the sum of the round trip and service time

Computational Results

Table 1 : Computational results for smaller problems with **$|N|$ < 40**. Drone parameters: L={0,1}, (B^0, α^0, β^0)=(100,0.4,0.4) and (B^1, α^1, β^1)=(50,0.2,0.2). *: time limit (3600 seconds) reached

Problem	$\|N\|$	Cplex				Benders				Time ratio	Drone ratio(%)
		Time	BnB	GAP(%)	Obj.	Time	BnB	GAP(%)	Obj.		
C	25	301.2	447 678	0.0	173.71	28.3	2225	0.0	173.71	10.63	84.0
R	25	52.8	74 325	0.0	360.81	17.5	1811	0.0	360.81	3.02	56.0
RC	25	3600.0*	1 745 751	16.2	277.67	37.1	6153	0.0	277.67	**>97.08**	80.0
A-n32-k5	31	430.6	428 257	0.0	575.71	92.3	14 046	0.0	575.71	4.67	51.6
A-n33-k5	32	3600.0*	1 572 781	2.8	549.29	1039.5	83 382	0.0	548.74	**>3.47**	50.0
A-n33-k6	32	3202.8	1 736 877	0.0	552.37	2891.2	64 337	0.0	552.37	1.11	50.0
A-n34-k5	33	3600.0*	2 106 201	5.7	580.98	3600.0*	63 508	1.6	580.57	-	51.5
A-n36-k5	35	3600.0*	1 400 170	1.6	602.55	3600.0*	39 569	1.2	602.55	-	45.7
A-n37-k5	36	873.9	628 443	0.0	633.74	3600.0*	174 774	1.9	633.74	<0.24	44.4
A-n37-k6	36	1149.7	880 493	0.0	635.22	3600.0*	53 406	0.0	635.22	<0.32	47.2
A-n38-k5	37	3119.6	1 762 775	0.0	615.36	3600.0*	394 592	0.4	615.36	<0.87	48.6
A-n39-k5	38	3600.0*	1 589 056	1.4	696.34	3600.0*	37 820	3.7	701.84	-	39.5
A-n39-k6	38	1586.9	823 580	0.0	686.81	3600.0*	65 320	2.3	686.81	<0.44	39.5
B-n31-k5	30	3527.8	1 491 121	0.0	366.19	188.0	11 497	0.0	366.19	18.76	80.0
B-n34-k5	33	3600.0*	2 320 989	16.9	418.52	309.0	18 489	0.0	416.11	**>11.66**	75.8
B-n35-k5	34	3600.0*	1 926 631	19.4	479.80	290.2	36 736	0.0	479.74	**>12.41**	73.5
B-n38-k6	37	3600.0*	2 056 881	16.9	465.87	1040.5	93 853	0.0	465.87	**>3.46**	73.0
B-n39-k5	38	3600.0*	2 334 460	21.0	458.87	3600.0*	52 557	15.5	459.49	-	71.1
P-n16-k8	15	2.5	5875	0.0	150.92	4.1	271	0.0	150.92	0.61	80.0
P-n19-k2	18	12.9	22 383	0.0	187.08	7.8	771	0.0	187.08	1.65	77.8
P-n20-k2	19	16.2	25 097	0.0	200.37	9.7	729	0.0	200.37	1.67	73.7
P-n21-k2	20	44.5	85 038	0.0	208.71	11.8	949	0.0	208.71	3.76	75.0
P-n22-k2	21	69.8	88 403	0.0	213.99	18.2	1809	0.0	213.99	3.83	76.2
P-n22-k8	21	402.5	662 730	0.0	310.94	13.6	1218	0.0	310.94	29.61	61.9
P-n23-k8	22	375.9	470 368	0.0	219.77	22.5	1128	0.0	219.77	16.74	77.3
P-n40-k5	39	1403.8	543 432	0.0	517.96	3600.0*	27 474	3.7	526.58	<0.39	43.6

The more clustered the nodes, the faster the Benders approach becomes compared to Cplex.

Computational Results

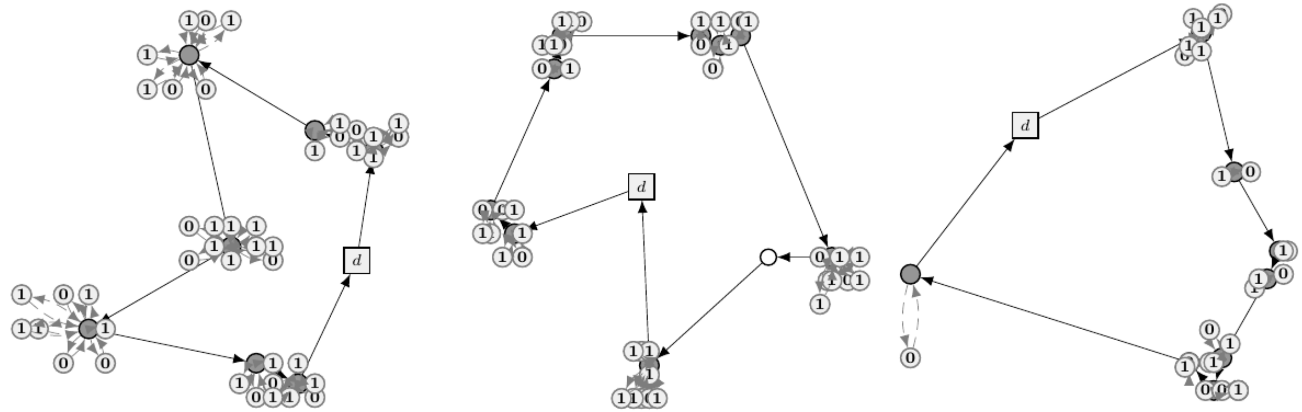
Table 2 : Computational results for smaller problems with **$|N| \geq 40$**. Drone parameters: L={0, 1}, (B^0, α^0, β^0)=(160,0.4,0.4) and (B^1, α^1, β^1)=(100,0.2,0.2). *: time limit (3600 seconds) reach ed

Problem	$\|N\|$	Cplex				Benders				Time ratio	Drone ratio(%)
		Time	BnB	GAP(%)	Obj.	Time	BnB	GAP(%)	Obj.		
A-n44-k7	43	3600.0*	1 763 367	6.9	625.24	3600.0*	165 414	0.7	619.55	-	65.1
A-n45-k6	44	3600.0*	1 555 005	8.8	663.68	3600.0*	96 557	4.9	666.50	-	59.1
A-n45-k7	44	3600.0*	1 733 985	5.7	594.33	3588.3	283 678	0.0	594.33	>1.00	63.6
A-n46-k7	45	3600.0*	1 515 087	5.6	615.56	2597.2	134 365	0.0	615.56	>1.39	62.2
A-n48-k7	47	3600.0*	972 128	5.0	627.99	3600.0*	155 447	1.5	626.42	-	63.8
A-n53-k7	52	3600.0*	1 266 221	8.4	728.30	3600.0*	242 514	1.9	715.42	-	59.6
A-n54-k7	53	3600.0*	1 010 292	6.3	699.73	3600.0*	19 724	6.9	700.23	-	62.3
A-n55-k9	54	3600.0*	981 884	4.9	688.75	3600.0*	66 334	4.6	695.02	-	63.0
A-n60-k9	59	3600.0*	1 288 335	8.4	752.59	3600.0*	59 023	6.3	756.23	-	61.0
A-n61-k9	60	3600.0*	964 437	7.2	732.36	3600.0*	16 196	7.7	732.19	-	60.0
B-n41-k6	40	3600.0*	1 760 307	25.5	440.49	3600.0*	123 626	4.2	435.26	-	82.5
B-n43-k6	42	3600.0*	944 089	9.6	404.10	3600.0*	67 159	2.3	404.98	-	78.6
B-n44-k7	43	3600.0*	1 702 750	27.5	403.58	3600.0*	190 100	10.2	395.59	-	83.7
B-n45-k5	44	3600.0*	1 453 467	20.5	497.34	3530.7	209 692	0.0	494.15	>1.02	79.5
B-n45-k6	44	3600.0*	1 603 296	12.4	402.25	3600.0*	126 720	1.3	398.74	-	86.4
B-n50-k7	49	3600.0*	1 495 864	26.8	483.53	3600.0*	169 600	13.0	480.36	-	81.6
B-n50-k8	49	3600.0*	1 361 341	24.8	534.30	3600.0*	18 443	7.1	536.55	-	75.5
B-n51-k7	50	3600.0*	1 463 453	33.0	550.90	3600.0*	151 400	24.7	552.93	-	82.0
B-n52-k7	51	3600.0*	1 167 144	30.6	481.37	3600.0*	241 640	20.5	479.70	-	82.4
B-n56-k7	55	3600.0*	1 727 802	36.3	497.97	3600.0*	81 062	21.8	501.58	-	81.8
B-n57-k9	56	3600.0*	679 088	17.4	587.73	3600.0*	139 241	7.0	589.62	-	76.8
P-n45-k5	44	3600.0*	1 278 188	1.1	456.76	1117.5	43 456	0.0	456.76	>3.22	72.7
P-n50-k7	49	3600.0*	1 134 859	2.7	473.37	3600.0*	64 411	0.8	471.09	-	71.4
P-n51-k10	50	3600.0*	1 034 370	6.0	524.74	3600.0*	51 007	3.5	518.48	-	66.0
P-n55-k7	54	3600.0*	953 710	4.5	527.98	3600.0*	29 437	6.5	533.98	-	66.7
P-n60-k10	59	3600.0*	538 820	5.1	604.53	3600.0*	22 420	7.0	602.15	-	62.7
P-n60-k15	59	3600.0*	533 798	4.5	604.59	3600.0*	24 991	6.7	608.80	-	61.0

Our algorithm found better incumbent solutions with much smaller GAP values for most of the cases.

Best-known solutions to selected problems.

The clustered nodes (a),(b),(c) : Each drone took multiple deliveries at a small number of waiting nodes.



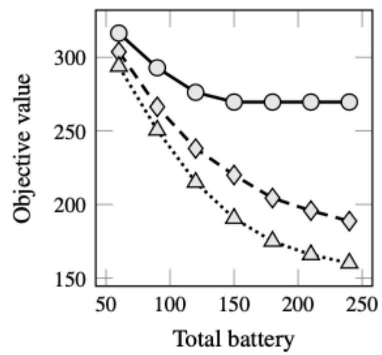
(a) C with 50 nodes (b) RC with 50 nodes (c) B-n34-k5

The distributed nodes (d),(e),(f) : Result in many waiting nodes with shorter waiting times at each waiting node.

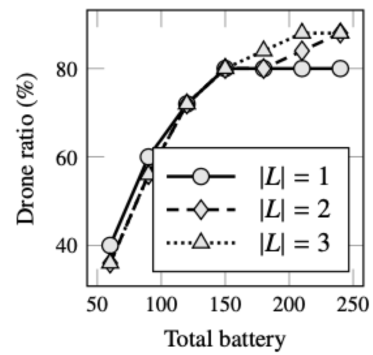
(d) R with 50 nodes (e) A-n39-k5 (f) P-n40-k5

Sensitivity Analysis

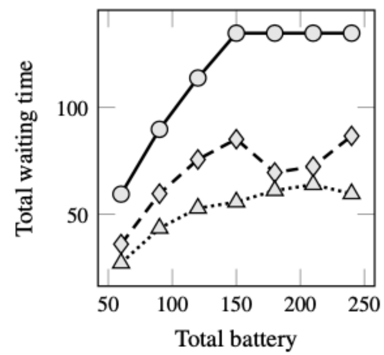
To assert the changes in the solutions for different drone configurations, we solved the same problems with various drone parameters. (Solomon C and R with 25 nodes)



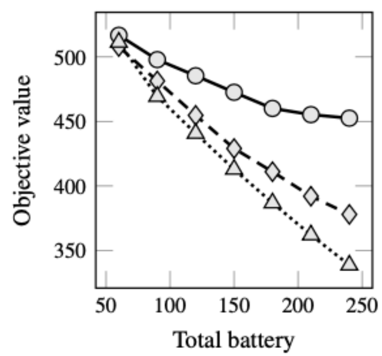
(a) Objective value for C



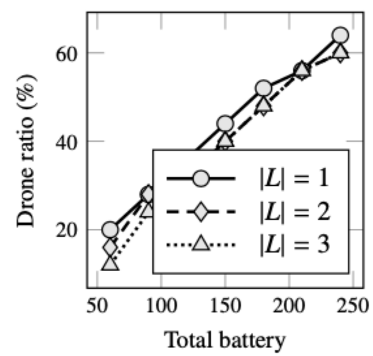
(b) Drone ratio for C



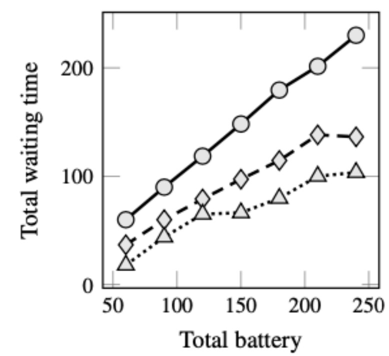
(c) Total waiting time for C



(d) Objective value for R



(e) Drone ratio for R

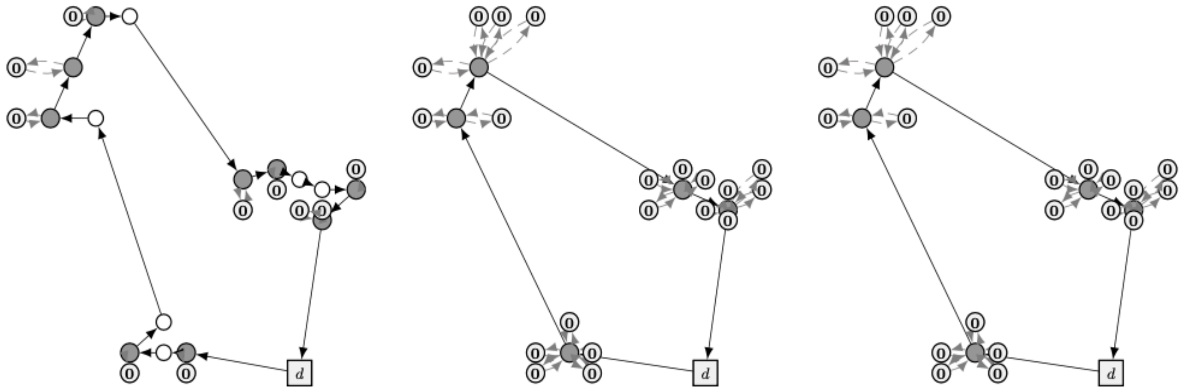


(f) Total waiting time for R

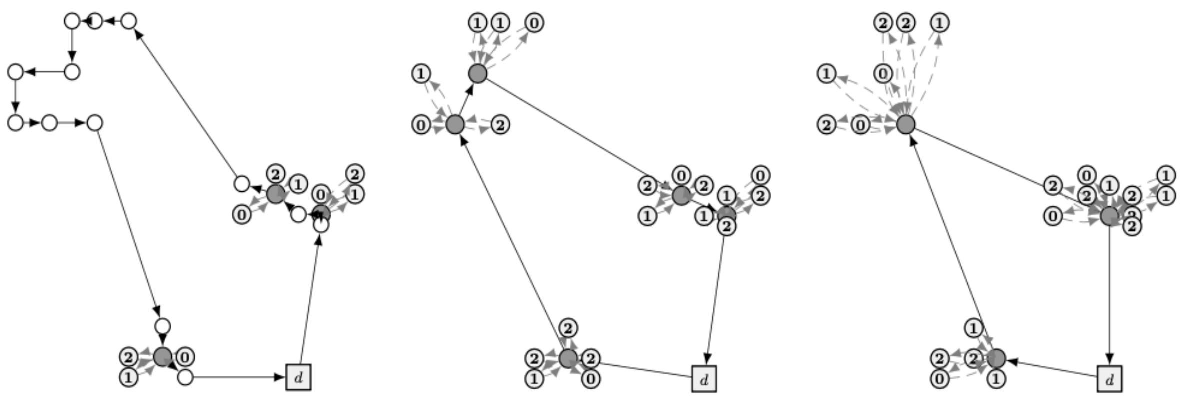
The objective values decrease faster for the multiple drone cases, while the drone ratio values increase at a similar rate for all cases. Also there is a possibility of further reduction of objective value by having more drone batteries.

Solutions for different drone configurations

Solomon C problem with 25 nodes. TB: Total battery.



(a) TB=60, $|L| = 1$ (b) TB=150, $|L| = 1$ (c) TB=240, $|L| = 1$

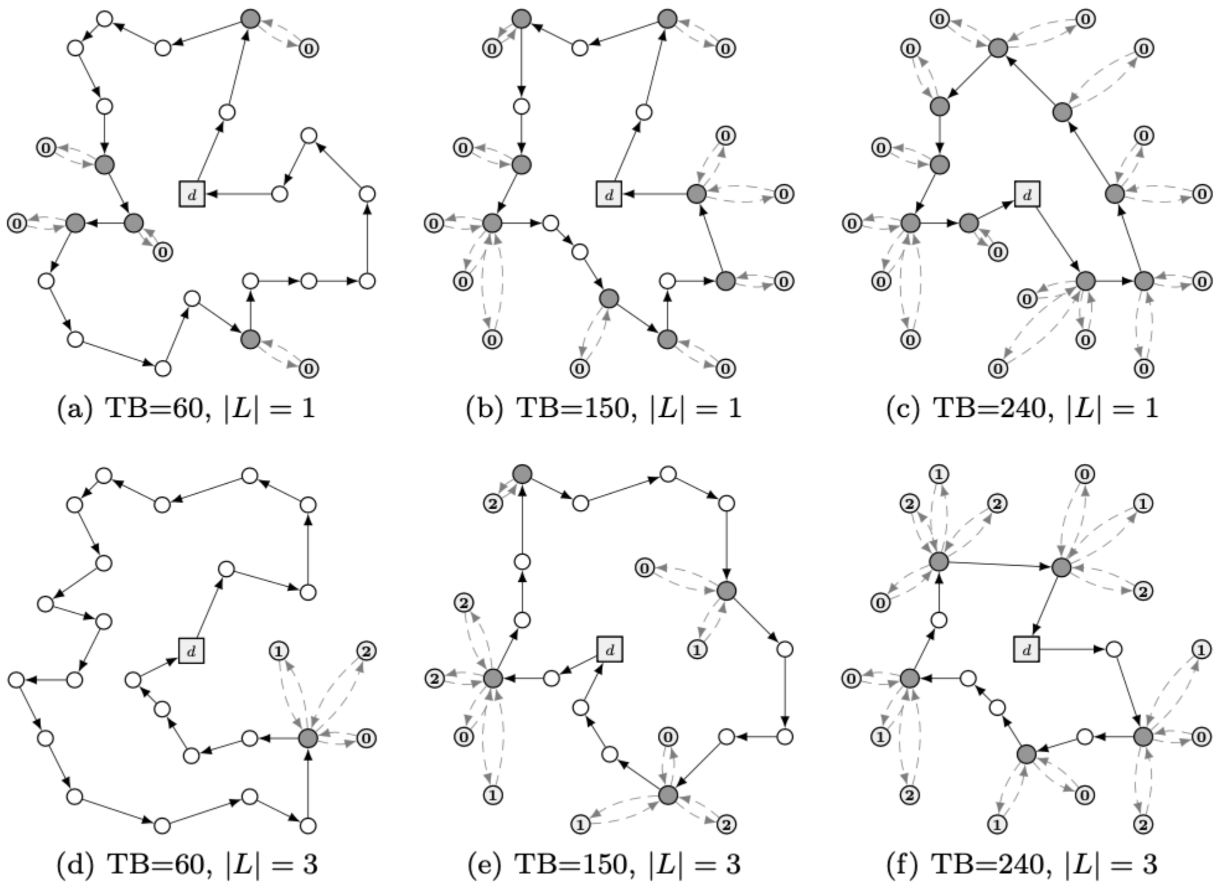


(d) TB=60, $|L| = 3$ (e) TB=150, $|L| = 3$ (f) TB=240, $|L| = 3$

The truck routes of solutions remain similar regardless of the drone parameters.
When there is a sufficiently large total drone battery, the truck visits a single node in each clustered area.

Solutions for different drone configurations

Solomon R problem with 25 nodes. TB: Total battery.



(a) TB=60, $|L| = 1$ (b) TB=150, $|L| = 1$ (c) TB=240, $|L| = 1$

(d) TB=60, $|L| = 3$ (e) TB=150, $|L| = 3$ (f) TB=240, $|L| = 3$

The truck routes undergo significant changes with different drone parameters because there are no apparent center nodes.

Conclusion

- We considered the case in which the heterogeneous drones are carried by truck and can be used for delivery while the truck is parked and waiting. (HDTRP)

- We presented a mathematical formulation for the problem.

- We developed an exact algorithm based on the logic-based Benders decomposition approach.

- To accelerate the proposed Benders algorithm, we also developed a set of preprocessing steps (primal heuristics, variable bounding)

- We reported an extensive computational study that shows our algorithm outperforms the state-of-the-art MIP solver.