Coordinating drones with mothership vehicles: The mothership and drone routing problem

Lavinia Amorosi Justo Puerto Carlos Valverde

September 2020

1 Description of the Problem

In the Mothership and Drone Routing Problem with Graphs (MDRPG), there is one mothership (the base vehicle) and one drone and the problem consists on the coordination between the drone and the base vehicle to minimize the total distance travelled by both vehicles. In this case, we assume that there exist no obstacles to prevent drone travel from moving in straight line segments.

The mothership and the drone begin at a starting location, denoted *orig*. There exists a set $\mathcal{G}$ of target locations modeled by graphs. For each stage $t \in \{1, \dots, |\mathcal{G}|\}$, we require that the drone is launched from the current mothership location, that at stage t is a decision variable denoted by x_L^t, flies to one of the graphs that has to be visited G, traverses G and then returns to the current position of the mothership that is another decision variable denoted by x_R^t. Once all targets graphs have been visited, the mothership and drone return to a final location (depot), denoted by *dest*.

Let $G = (V_G, E_G)$ be a graph in $\mathcal{G}$ and (g, i) be the pair that denotes the edge i of this graph G. This edge is parameterized by its endpoints B_g^i, C_g^i and its length $\|\overline{B_g^i C_g^i}\|$ is denoted by $\mathcal{L}(g, i)$. For each line segment, we assign a binary variable μ_g^i that indicates whether the drone visits the segment (g, i) and define entry and exit points (R_g^i, ρ_g^i) and (L_g^i, λ_g^i), respectively, that model the part of the edge visited by the drone.

We have considered two modes of visit to the targets graphs G:

- Visiting a percentage α_g^i of each edge (g, i) which can be modeled by using the following constraints:

$$\mu_g^i |\lambda_g^i - \rho_g^i| \geq \alpha_g^i, \quad \forall g, i \tag{α-E}$$

- Visiting a percentage α_g of the total length of the graph:

$$\sum_i \mu_g^i |\lambda_g^i - \rho_g^i| \mathcal{L}(g, i) \geq \alpha_g \mathcal{L}(g), \quad \forall g \tag{α-G}$$

  where $\mathcal{L}(g)$ denotes the total length of the graph.

In both cases, we need to introduce a binary variable that determines the traveling direction on the edge (g, i) as well as the definition of the parameter values $\nu_g^{\text{i, min}}$ and $\nu_g^{\text{i, max}}$ of the access and exit points to that segment. Then, for each edge (g, i), the absolute value restriction can be represented by:

$$\mu_g^i |\rho_g^i - \lambda_g^i| \geq \alpha_g^i \Longleftrightarrow \left\{ \begin{array}{ccl} \rho_g^i - \lambda_g^i & = & \nu_g^{\text{max, i}} - \nu_g^{\text{min, i}} \\ \nu_g^{\text{max, i}} & \leq & 1 - \text{entry}_g^i \\ \nu_g^{\text{min, i}} & \leq & \text{entry}_g^i, \\ \mu_g^i (\nu_g^{\text{max, i}} + \nu_g^{\text{min, i}}) & \geq & \alpha_g^i \end{array} \right. \tag{α-E}$$

The linearization of (**??**) is similar to (**??**) by changing the last inequality in (**??**) for

$$\sum_i \mu_g^i (\nu_g^{\text{max, i}} + \nu_g^{\text{min, i}}) \mathcal{L}(g, i) \geq \alpha_g \mathcal{L}(g). \tag{α-G}$$

The mothership and drone do not need to arrive at a rendezvous location at the same time; the faster arriving vehicle may wait for the other at the rendezvous location.

We assume that vehicles move at constant speeds, although this hypothesis could be relaxed. The mothership travels at v_M speed whereas the drone has a speed of $v_D > v_M$. The mothership and the drone must travel together from *orig* to the first launch location. Similarly, after the drone visits the last target location, the mothership and the drone must meet at the final rendezvous location before traveling together back to *dest*. The first launch location and final rendezvous location are allowed to be *orig* and *dest*, respectively, but it is not mandatory. For the ease of presentation, in this paper we will assume that *orig* and *dest* are the same location. However, all results are extend easily to the case that *orig* and *dest* are different locations.

The goal is to find a minimum time path that begins at *orig*, ends at *dest*, and where every $G \in \mathcal{G}$ is visited by the drone.

Depending on the assumptions made on the movements of the mothership vehicle this problem gives rise to two different versions: a) the mothership vehicle can move freely on the continuous space (all terrain ground vehicle, boat on the water or flying vehicle); and b) the mothership vehicle must move on a travel network (that is, it is a normal truck or van). In the former case, that we will call All terrain Mothership-Drone Routing Problem with Graphs (AMDRPG), each launch and rendezvous location may be chosen from a continuous space (the Euclidean 2-or-3 dimension space). In the latter case, that we will call Network Mothership-Drone Routing Problem with Graphs (NMDRPG) from now on, each launch and rendezvouz location must be chosen on a given graph embedded in the considered space.

2 Mixed Integer Non Linear Programming Formulations

In this section we present alternative MINLP formulations for the (MDRPG) depending on the nature of the mothership that will be compared computationally in later sections. We start analyzing first the simplest model, namely the (AMDRPG). We shall address later the second model (NMDRPG). As it will clear later, we will elaborate from the previous model modifying some intrinsic conditions that characterize each problem.

By using the notation below, in each stage $t \in \{1, \ldots, |\mathcal{G}|\}$, the drone makes the following path:

$$x_L^t \rightarrow R_g^i \rightarrow L_g^i \rightarrow \ldots \rightarrow R_g^j \rightarrow L_g^j \rightarrow \ldots \rightarrow R_g^u \rightarrow x_R^t \rightarrow x_L^{t+1}.$$

This path enforces the definition of binary variables that choose:

- The optimal order to visit each graph $G \in \mathcal{G}$.
- The optimal order to visit the edges of each graph.

Therefore, it is necessary to define the following binary variables:

- $u_g^{it} = 1$ if the drone enters the graph g by the segment (g, i) at the stage t.
- $z_g^{ij} = 1$ if the drone goes from the segment (g, i) to the segment (g, j).
- $v_g^{it} = 1$ if the drone exits the graph by the segment (g, i) at the stage t.

By using these binary variables, we can model the route that follows the drone:

$$\sum_{i,g} u_g^{it} = 1, \qquad \forall t \tag{1}$$

$$\sum_{i,g} v_g^{it} = 1, \qquad \forall t \tag{2}$$

$$\sum_{i,t} u_g^{it} = 1, \qquad \forall g \tag{3}$$

$$\sum_{i,t} v_g^{it} = 1, \qquad \forall g \tag{4}$$

$$\sum_{j} z_g^{ji} + \sum_{t} u_g^{it} = \mu_g^i, \qquad \forall g, i \tag{5}$$

$$\sum_{j} z_g^{ij} + \sum_{t} v_g^{it} = \mu_g^i, \qquad \forall g, i \tag{6}$$

Equations (**??**) and (**??**) state that in each stage we visit (enter and exit, respectively) only one graph. Constraints (**??**) and (**??**) assure that each graph is visited in some stage. Constraints (**??**) (resp. (**??**)) say that the number of exterior edges plus the number of interior edges that enter (resp. exit) to the segment (g, i) is given by μ_g^i.

Elimination of subtours

To prevent the existence of subtours within each graph that the drone must visit, one can include either the compact formulation that uses the Miller-Tucker-Zemlin constraints or the subtour elimination constraints (SEC).

For the MTZ formulation, we define the continuous variables s_g^i that indicate the order to visit the edge (g, i) and establish the following constraints:

$$s_g^i - s_g^j + |E_g| z_g^{ij} \leq |E_g| - 1, \qquad \forall g, i \neq j \qquad (\mathrm{MTZ}_1)$$

$$0 \leq s_g^i \leq |E_g| - 1 \qquad \forall g, i \qquad (\mathrm{MTZ}_2)$$

On the other hand, we can also use the family of subtour elimination constraints:

$$\sum_{i,j \in S} z_g^{ij} \leq |S| - 1, \quad \forall g \in \mathcal{G}, \quad \forall S \subset E_g \qquad (\mathrm{SEC})$$

(Copiado del XPPN) Since there is an exponential number of SEC constraints, when we implement this formulation we need to perform a row generation procedure including constraints whenever they are required by a separation oracle. To find SEC inequalities, as usual, we search for disconnected components in the current solution. Among them, we choose the shortest subtour found in the solution to be added as a lazy constraint to the model.

2.1 All terrain Mothership-Drone Routing Problem with Graphs

In this problem, we assume that the mothership is an aircraft so that it is allowed to move freely in $\mathbb{R}^2$. The goal of the (AMDRPG) is to find a feasible solution that minimizes the total distance traveled by the drone and the mothership. Hence, assuming that lengths are given by the Euclidean distance, $\| \cdot \|_2$, between their endpoints, we need to define the following distance variables:

- $d_g^{Lit} = \|x_L^t - R_g^i\|$. Distance traveled by the drone from the launch point at the stage t to the first visiting point in the segment (g, i) of the graph g.
- $d_g^{ij} = \|R_g^i - L_g^j\|$. Distance traveled by the drone from the launch point in (g, i) to the rendezvous point in (g, j).
- $d_g^i = \|R_g^i - L_g^i\|$. Distance traveled by the drone from the retrieve point to the next launch point in (g, i).
- $d_g^{Rit} = \|L_g^i - x_R^t\|$. Distance traveled by the drone from the launch point in the segment (g, i) to the retrieve point on the mothership at the stage t.
- $d^{LRt} = \|x_L^t - x_R^t\|$. Distance traveled by the mothership from the launch point to the retrieve point at the stage t.
- $d^{RLt} = \|x_R^t - x_L^{t+1}\|$. Distance traveled by the mothership from the retrieve point in the stage t to the launch point in the stage $t + 1$.

2.1.1 A first formulation for AMDRPG based on stages

$$
\begin{aligned}
\min \quad & \sum_{i,g,t}(u_g^{it}d_g^{Lit} + v_g^{it}d_g^{Rit}) + \sum_{g,i} p_g^i \mathcal{L}(g,i) + \sum_{i,j,g} z_g^{ij} d_g^{ij} + \sum_t (d^{RLt} + d^{LRt}) && \text{(AMDRPG)}\\
\text{s.t.} \quad & (??) - (??), \\
& (??) - (??) \text{ or } (??), \\
& (??) \text{ or } (??), \\
& \|x_L^t - R_g^i\| \leq d_g^{Lit}, && \forall g,i,t,\\
& \|R_g^i - L_g^i\| \leq d_g^i, && \forall g,i,t,\\
& \|R_g^i - L_g^j\| \leq d_g^{ij}, && \forall g,i,j,\\
& \|L_g^i - x_R^i\| \leq d_g^{Rit}, && \forall g,i,t,\\
& \|x_R^t - x_L^{t+1}\| \leq d_g^{RLt}, && \forall g,i,t,\\
& \|x_L^t - x_R^t\| \leq d_g^{LRt}, && \forall g,i,t,\\
& \left(\sum_t u_g^{it}d_g^{Lit} + \sum_{i,j} z_g^{ij}d_g^{ij} + \sum_i d_g^i + \sum_t v_g^{it}d_g^{Rit}\right)/v_D \leq d^{RLg}/v_M, && \forall g,i, \qquad \text{(DCW)}\\
& x_L^0 = orig,\\
& x_R^0 = orig,\\
& x_L^{|\mathcal{G}|+1} = dest,\\
& x_R^{|\mathcal{G}|+1} = dest.
\end{aligned}
$$

Constraints (??) ensure that the time spent by the drone to visit the graph g at the stage t is less than or equal to the time that the mothership needs to move from the launch point to the retrieve point at the stage t. If we want to penalized the waiting time of the drone, we can add a slack variable in the restriction that has to be minimized in the objective function.

Observe that we are assuming constant velocities for the mothership v_M and the drone v_D.

To deal with the bilinear terms of the objective function, we use McCormick's envelopes to linearize these terms by adding variables $p \geq 0$ that represent these products and introduce these constraints:

$$
\begin{aligned}
p &\geq mz,\\
p &\leq d - M(1-z),
\end{aligned}
$$

where m and M are, respectively, the lower and upper bounds of the distance variable d. These bounds will be adjusted for each bilinear term in Section **??**.

2.1.2 Alternative formulation for AMDRPG based on MTZ-constraints

This path enforces the definition of binary variables that choose:

- The optimal order to visit each graph $G \in \mathcal{G}$.
- The optimal order to visit the edges of each graph.

Therefore, it is necessary to define the following binary variables:

- $u_g^i = 1$ if the drone enters the graph g by the segment (g,i).
- $z_g^{ij} = 1$ if the drone goes from the segment (g,i) to the segment (g,j).
- $v_g^i = 1$ if the drone exits the graph by the segment (g,i).
- $w_{gg'} = 1$ if the truck goes from x_L^g to $x_R^{g'}$.

By using these binary variables, we can model the route that follows the drone:

$$\sum_i u_g^i = 1, \qquad \forall g \tag{7}$$

$$\sum_i v_g^i = 1, \qquad \forall g \tag{8}$$

$$\sum_j z_g^{ji} + u_g^i = \mu_g^i, \qquad \forall g, i \tag{9}$$

$$\sum_j z_g^{ij} + v_g^i = \mu_g^i, \qquad \forall g, i \tag{10}$$

On the other hand, to model the tour of the truck, we include these constraints:

$$\sum_{g \in \mathcal{G}} w_{g0} = 0, \tag{11}$$

$$\sum_{g' \in \mathcal{G}} w_{(n_G+1)g'} = 0, \tag{12}$$

$$\sum_{g' \in \mathcal{G}' \setminus \{g\}} w_{gg'} = 1, \qquad \forall g \in \mathcal{G} \tag{13}$$

$$\sum_{g \in \mathcal{G}' \setminus \{g'\}} w_{gg'} = 1, \qquad \forall g' \in \mathcal{G} \tag{14}$$

$$s_g - s_{g'} + |\mathcal{G}'| w_{gg'} \leq |\mathcal{G}'| - 1, \qquad \forall g \neq g' \tag{MTZ$_3$}$$

$$0 \leq s_g \leq |\mathcal{G}'| - 1 \qquad \forall g \in \mathcal{G} \tag{MTZ$_4$}$$

$$s_0 = 0, \tag{MTZ$_5$}$$

$$s_{n_G+1} = n_G + 1, \tag{MTZ$_6$}$$

In this problem, we assume that the mothership is an aircraft so that it is allowed to move freely in $\mathbb{R}^2$. Hence, assuming that lengths are given by the Euclidean distance, $\|\cdot\|_2$, between their endpoints, we need to define the following distance variables:

- $d_g^{Li} = \|x_L^g - R_g^i\|$. Distance traveled by the drone from the launch point to the first visiting point in the segment (g, i) of the graph g.
- $d_g^{ij} = \|R_g^i - L_g^j\|$. Distance traveled by the drone from the launch point in (g, i) to the rendezvous point in (g, j).
- $d_g^i = \|R_g^i - L_g^i\|$. Distance traveled by the drone from the retrieve point to the next launch point in (g, i).
- $d_g^{Ri} = \|L_g^i - x_R^g\|$. Distance traveled by the drone from the launch point in the segment (g, i) to the retrieve point on the mothership.
- $d_g^{LR} = \|x_L^g - x_R^g\|$. Distance traveled by the mothership from the launch point to the retrieve point while the drone is visiting g.
- $d_{gg'}^{RL} = \|x_R^g - x_L^{g'}\|$. Distance traveled by the mothership from the retrieve point for the graph g to the launch point associated to the graph g'.

The goal of the (AMDRPG) is to find a feasible solution that minimizes the total distance traveled by the drone and the mothership, i.e.,

$$
\begin{array}{llr}
\min & \sum_{i,g}(u_g^i d_g^{Li} + v_g^i d_g^{Ri}) + \sum_{g,i} p_g^i \mathcal{L}(g,i) + \sum_{g,i,j} z_g^{ij} d_g^{ij} + \sum_{g} d_g^{LR} + \sum_{g,g'} d_{gg'}^{RL} w_{gg'} & \text{(AMDRPG)} \\
\text{s.t.} & (??) - (??), & \\
& (??) - (??) \text{ or } (??), & \\
& (??) - (??), & \\
& (??) - (??) \text{ or } (??), & \\
& (??) \text{ or } (??), & \\
& \|x_L^g - R_g^i\| \leq d_g^{Li}, \quad \forall g, i, & \\
& \|R_g^i - L_g^i\| \leq d_g^i, \quad \forall g, i, & \\
& \|R_g^i - L_g^j\| \leq d_g^{ij}, \quad \forall g, i, j, & \\
& \|L_g^i - x_R^i\| \leq d_g^{Ri}, \quad \forall g, i, & \\
& \|x_R^g - x_L^{g'}\| \leq d_{gg'}^{RL}, \quad \forall g, g', & \\
& \|x_L^g - x_R^g\| \leq d_g^{LR}, \quad \forall g, & \\
& \left(u_g^i d_g^{Li} + \sum_{i,j} z_g^{ij} d_g^{ij} + \sum_{i} p_g^i \mathcal{L}(g,i) + v_g^i d_g^{Ri}\right) / v_D \leq d_g^{RL}/v_M, \quad \forall g, i, & \text{(DCW)} \\
& x_L^0 = orig, & \\
& x_R^0 = orig, & \\
& x_L^{|\mathcal{G}|+1} = dest, & \\
& x_R^{|\mathcal{G}|+1} = dest. &
\end{array}
$$

3 Strengthening the formulation of XPPN

The different models that we have proposed include in one way or another big-M constants. We have defined different big-M constants along this work. In order to strengthen the formulations we provide good upper bounds and lower bounds for those constants. In this section we present some results that adjust them for each kind of set considered in our models.

Distance from the launch / retrieve point in the mothership to the retrieve / launch point in the graph

- Aircraft case. When the mothership is an aircraft, the best upper bound M_g^{Rit} or M_g^{Lit} that we can consider is the distance between the *orig* and the furthest point from the *orig*, i.e., every launch or retrieve point is inside the circle whose diametrically opposite points are the explained before:

$$M_g^{Rit} = \max_{\{v \in V_G : G \in \mathcal{G}\}} \|orig - v\| = M_g^{Lit}.$$

  On the other hand, the minimum distance in this case can be zero, that occurs when the launch or the retrieve point of the aircraft is the same to the retrieve or the launch point in the graph.

- Truck case. In this case, the best upper bound M_g^{Rit} or M_g^{Lit} is the maximum distance between the polygonal chain and the graphs:

$$M_g^{Rit} = \max_{\{v \in V_G : G \in \mathcal{G}\}} \max_{w \in \mathcal{P}} \|v - w\| = M_g^{Lit}.$$

  On the other hand, the minimum distance can be computed by taking the closest points between the graphs and the polygonal chain:

$$m_g^{Rit} = \min_{\{v \in V_G : G \in \mathcal{G}\}} \min_{w \in \mathcal{P}} \|v - w\| = m_g^{Lit}.$$

Distance from the launch point to the retrieve point in the graph

Aircraft / Truck case. Since we are taking to account the distance between two edges i and j, the maximum and minimum distances between their vertices give us the upper and lower bounds:

$$\begin{aligned} M_g^{ij} &= \max\{\|B_g^i - C_g^j\|, \|B_g^i - B_g^j\|, \|C_g^i - B_g^j\|, \|C_g^i - C_g^j\|\}, \\ m_g^{ij} &= \min\{\|B_g^i - C_g^j\|, \|B_g^i - B_g^j\|, \|C_g^i - B_g^j\|, \|C_g^i - C_g^j\|\}. \end{aligned}$$

Distance inside the polygonal for the truck case

An upper bound for the distance inside the polygonal is to take the total length of the line segments where the truck is located:

$$M_{ij}^t = \begin{cases} \mathcal{L}(i), & \text{if } i = j, \\ \sum_{k=i}^{j} \mathcal{L}(k) & \text{if } i < j. \end{cases}$$