

A Mixed Integer Linear Programming Model for the Drone-Assisted TSP

Joao Guilherme Cavalcanti Costa

Instituto de Ciencias Matemáticas e de Computação – Universidade de São Paulo – Brasil
Avenida Trabalhador São-carlense, 400 - Centro- São Carlos
joaogcavalcanti@usp.br

Allan Vinícius Silva

Instituto de Ciencias Matemáticas e de Computação – Universidade de São Paulo – Brasil
Avenida Trabalhador São-carlense, 400 - Centro- São Carlos
allan.vinicius.silva@usp.br

Matheus Augusto Fernandes de Assunção

Instituto de Ciencias Matemáticas e de Computação – Universidade de São Paulo – Brasil
Avenida Trabalhador São-carlense, 400 - Centro- São Carlos
matheusfernandes@usp.br

Maristela Oliveira dos Santos

Instituto de Ciencias Matemáticas e de Computação – Universidade de São Paulo – Brasil
Avenida Trabalhador São-carlense, 400 - Centro- São Carlos
mari@icmc.usp.br

ABSTRACT

In this article, the Drone-Assisted Traveling Salesman Problem (DATSP), a new context in which drones are used to extend a classic TSP, is explored. Drones travel unimpeded by traffic and can reach areas of difficult access, whereas other vehicles might not be able to do so. In this new paradigm, an obstacle, such as a river or a mountain, would hinder a company's delivery time to the other side of that obstacle. The drone would then fly repeatedly across it, carrying packages which are then delivered to the clients by a deliveryman. This research is motivated by a real life application, where an Icelandic company started using drones to deliver food across a bay. This operational problem inspired the DATSP, which is modeled mathematically as a Mixed Integer Linear Programming (MILP) problem in this article. Finally, instances were generated and solved with the solver GUROBI and their solutions were investigated.

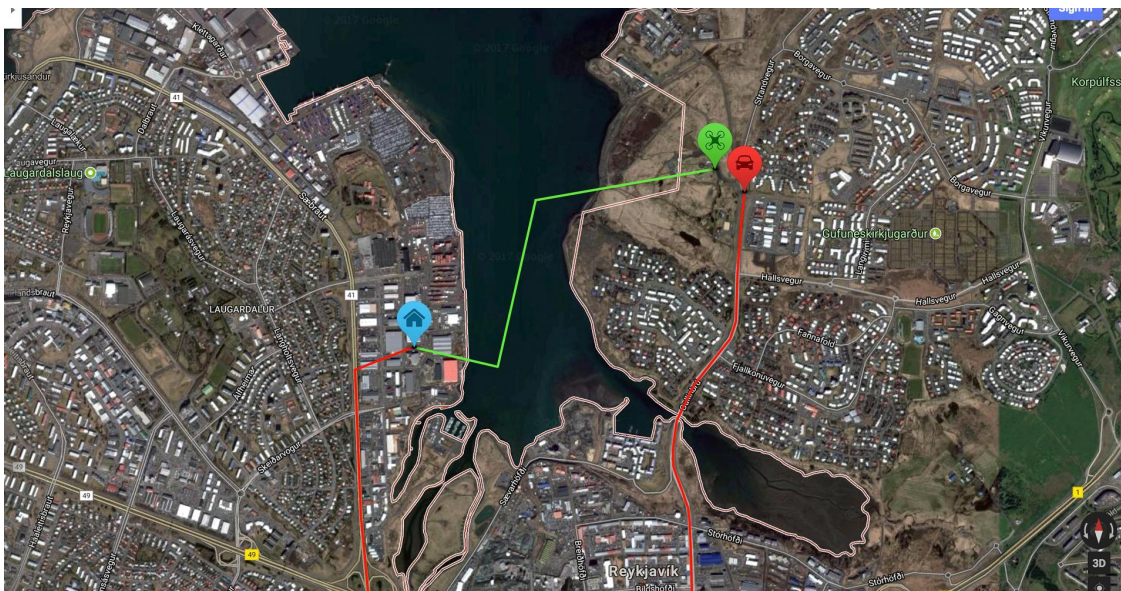
KEYWORDS. Drones. Traveling Salesman Problem. Mathematical Model.

Paper topics (L&T - Logística e Transportes)

1. Introduction

Drones have increasingly become part of modern society. Many companies have sought to streamline their services using drones in the context of delivering goods to customers. Icelandic company AHA, for instance, is already using Flytrex's drones to fly over a bay in Reykjavik, cutting costs and delivery times significantly [Gilchrist, 2017], as seen in Figure 1. Zipline, a company in Silicon Valley, has teamed up with the Rwandan health department and is saving lives using drones to deliver blood and medicine to hospitals in remote regions [Simmons, 2016]. DHL, the German postal service, trialled delivery drones in combination with fixed stations between which the drones fly, saving around 20 minutes in delivery time during winter [DHL, 2016]. Amazon announced plans in 2013 to make deliveries using drones 30 minutes after customers purchase an item on their website [CBS, 2013]. Dominos is testing pizza deliveries using drones in New Zealand directly to customers [Scholz, 2016]. UPS successfully tested using drones in tandem with a truck to deliver products to clients in rural areas [Lithia, 2017].

Figure 1: A comparison between a drone's route (green) and a delivery vehicle's (red) in Reykjavik.



Source: Flytrex, found in [Gilchrist, 2017].

This sudden interest in this type of application is due to the advantages drones provide, such as not following traffic, being able to reach areas of difficult access, being efficient energy-wise [Stolaroff et al., 2018], and being greener alternatives, which could reduce greenhouse gas emissions by almost 54% [Stolaroff and Samaras, 2018] when compared to trucks. Due to these factors, drones are already being studied in the academic field. For instance, [Murray and Chu, 2015] use a drone in tandem with a truck in the context of last-mile delivery, similarly to what has been tested by UPS. The authors propose the use of a drone to deliver to clients while the truck delivers to other clients and serves as a base for the drone, reducing the total delivery time when compared to a classic Traveling Salesman Problem (TSP) model. The work of [Dorling et al., 2017] uses drones to make deliveries within a radius around the depot on their own, allowing for multiple trips and multiple clients in the same route.

Despite the benefits granted from using drones, there are still many hindrances that need to be assessed. The issue of privacy is among the problems being discussed, since drones use cameras for their guidance systems. The risk of damage to private property or to people in case of accidents is also taken into account. The re-use of drones and the toll on battery life are also issues

that hinder the continuous use of drones in the context of delivering goods, since modern delivery drone technology does not allow for long flight times. Moreover, modern cargo drone models have load capacities limited to only a few kilos. In spite of these issues, the authors deem pertinent the use of drones in logistics, since there are many potential benefits to using drones. Amazon's CEO, for instance, claimed in an interview that 86% of all deliveries the company makes weigh less than 3 kilos [Rose, 2015], which would more than justify their investments in drone technology.

This paper proposes and solves the Drone-Assisted Traveling Salesman Problem (DATSP), where a drone is employed to gradually transport products (such as food, medicine, or any products light enough) to a deliveryman, aiming to minimize the operational time. Unlike the other works mentioned before, the drone does not itself deliver to the customers, its function is to transport those products to the deliveryman who is then responsible for delivering them to the customers. In fact, the authors could not find a similar problem that was explored in the literature.

The rest of this paper is organized as follows: Section 2 explains the DATSP; Section 3 presents the mathematical model based on Mixed Integer Linear Programming techniques; Next, Section 4 shows the computational results obtained using the solver GUROBI. Finally, Section 5 concludes the work and presents research perspectives.

2. The Drone-Assisted TSP

The problem to be discussed in this section is, to the authors' best knowledge, not one that has been treated in the literature. The goal of the Drone-Assisted Traveling Salesman Problem (DATSP) is to utilize a drone to help in the delivery of goods by flying over areas of difficult access, such as rivers, lakes, bays, forests or mountains. The scenario involved comprises a drone, the obstacle, and the deliveryman waiting across from the obstacle. The drone flies over and drops off the goods to the deliveryman who delivers them to the clients on the other side. Since the total cargo to be delivered exceeds the drone's load capacity, multiple trips are necessary until every product has been delivered. After the drone unloads the products, the deliveryman executes the route and meets the drone at the drop-off point to receive a new batch of goods that needs to be delivered. The deliveryman's route is described by the solution to a TSP. Thus, the decisions of the DATSP are determining how many visits the drone makes, which client is served in which visit, and the deliveryman's routes, minimizing total operational time. A similar mode of operation being used by Icelandic company AHA demonstrates that using drones to help deliveries manages to reduce time and costs to the company [Gilchrist, 2017].

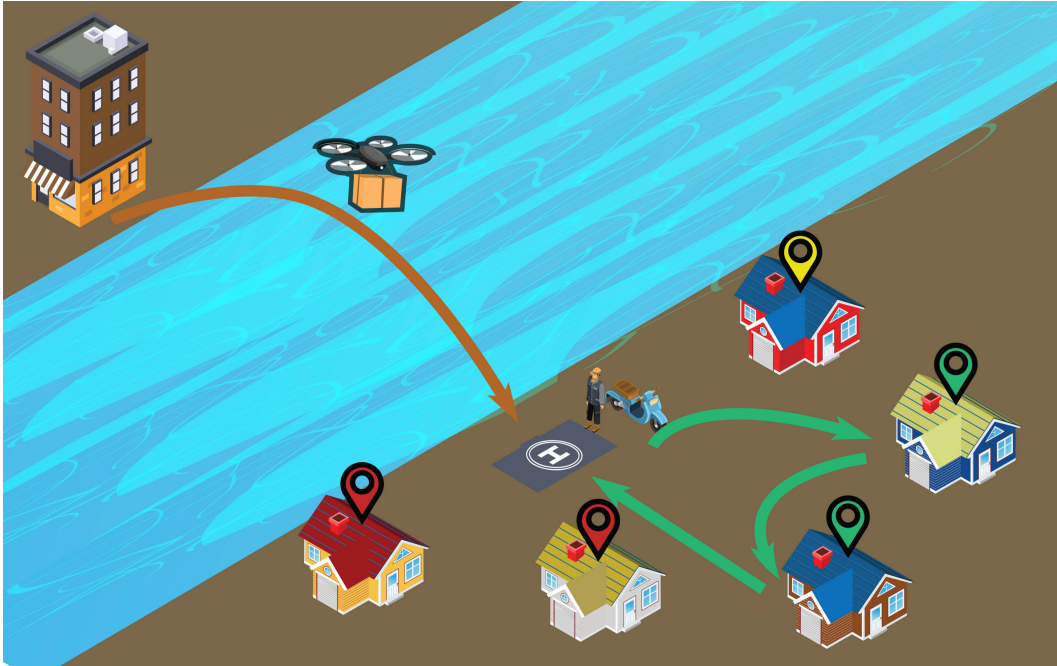
Figure 2 illustrates the DATSP, where two landmasses separated by a body of water can only be crossed using a boat or a ferry (there is no bridge nearby). On one side there is a company that delivers products locally. On the other side there are customers to whom delivering products might not be feasible without an alternative mode of transportation. The presence of an obstacle hence limits the number of possible customers to that company or discourages clients from using their services. Therefore, a drone can fly across the water and the deliveryman waiting on the other side can deliver to those customers. In the example, the route shown visits both customers with a green mark, whereas the other ones will be served during later visits.

Companies that deliver food, medicine or other light objects would benefit from this type of application by increasing their client base, reducing delivery times, improving market competitiveness and customer satisfaction. Given that this type of delivery is time-sensitive, the DATSP model could be used to schedule deliveries in short periods of time. For instance, in the context of food delivery, the customer does not expect to wait more than two hours, thus a possible strategy could be to solve the DATSP hourly or whenever a certain amount of orders is placed.

3. The Mathematical Model for the DATSP

The problem can be represented by a graph in which $V = \{0, 1, \dots, n\}$ is the set of vertices (0 being the drone's drop-off point) and $E = \{(i, j) | i, j \in V\}$ is the arcs set, in which (i, j) is an

Figure 2: An example of a DATSP application: A company utilizes the drone to cross the river.



Source: Authors, based on "Designed by Freepik" artwork.

arc that represents the deliveryman's transit from i to j . Each client i also has a demand that must be served by the deliveryman.

The main decisions of the DATSP are determining the number of trips the drone makes, the clients that deliveryman serves after each trip and the routes the deliveryman travels, aiming to minimize the total operational time. All demands are known at the start of the operation, and, since the total weight of the demands exceeds the drone's load capacity, many trips between the depot and the drone's drop-off point are necessary. For every time the drone unloads goods at the drop-off point, the route that must be executed by the deliveryman must be determined in a way that the deliveryman will always return to the drop-off point before the drone arrives with more products to be delivered. There is no priority between clients. The mathematical model for the DATSP is presented with its parameters and variables as follows.

Parameters:

- V : vertex set;
- C : client set ($C = V \setminus \{0\}$);
- M : set of the possible number of times the drone unloads at the drop-off point;
- l_i : the demand of client i ;
- d_{ij} : the distance between clients i and j ;
- Q_d : the drone's load capacity;
- Q_e : the deliveryman's load capacity;
- v_d : the drone's average flight speed;
- v_e : the deliveryman's average speed;
- τ : average total time the drone takes to take off and land;
- h : the distance between the depot and the drone's drop-off point;
- T_{max} : maximum operational time;
- K : a large enough number.

Variables:

x_{ijm} :	$\begin{cases} 1 & \text{if the deliveryman leaves } i \text{ and goes to } j \text{ on the drone's } m\text{-th visit;} \\ 0 & \text{otherwise.} \end{cases}$
p_{im} :	$\begin{cases} 1 & \text{if client } i \text{ is served on the drone's } m\text{-th visit;} \\ 0 & \text{otherwise.} \end{cases}$
s_{im} :	$\begin{cases} 1 & \text{if client } i \text{ has already been served on the drone's } m\text{-th visit or before;} \\ 0 & \text{otherwise.} \end{cases}$
y_m :	$\begin{cases} 1 & \text{if an } m\text{-th drone visit happens;} \\ 0 & \text{otherwise.} \end{cases}$
t_i :	instant in which the deliveryman leaves client i ;
a_{im} :	instant in which the deliveryman returns to the depot on the drone's m -th visit if client i was the last client in this particular visit;
T :	instant in which the deliveryman returns to the drop-off point after serving the last client overall;
γ_m :	instant in which the drone makes its m -th visit;
q_i :	total load delivered by the deliveryman after leaving client i .

- Flow constraints

$$\sum_{m \in M} p_{im} = 1, \quad \forall i \in C; \quad (1)$$

$$\sum_{\substack{i \in V \\ i \neq j}} x_{ijm} = p_{jm}, \quad \forall j \in C, \forall m \in M; \quad (2)$$

$$\sum_{j \in C} x_{0jm} = y_m, \quad \forall m \in M; \quad (3)$$

$$\sum_{\substack{i \in V \\ i \neq j}} x_{ijm} = \sum_{\substack{i \in V \\ i \neq j}} x_{jim}, \quad \forall j \in V, \forall m \in M; \quad (4)$$

Constraints (1) guarantee that every client is served by the end of the operation. Constraints (2) link the visit variables with the flow variables, where if a client j is served on the m -th visit, then the deliveryman necessarily serves j exactly once on that route. Constraints (3) demand that deliveries only happen in the m -th visit if and only if the drone makes that visit. Constraints (4) are responsible for ensuring the flow of the routes, meaning if the deliveryman arrives at client j 's location on the m -th visit, then he must also leave j on that same visit.

- Visit constraints

$$y_{m-1} \geq y_m, \quad \forall m \in M \setminus \{1\}; \quad (5)$$

$$p_{im} \leq s_{im}, \quad \forall i \in C, \forall m \in M; \quad (6)$$

$$s_{i,(m-1)} \leq s_{im}, \quad \forall j \in C, \forall m \in M \setminus \{1\}; \quad (7)$$

$$\sum_{i \in C} (1 - s_{i,(m-1)}) \leq |C| y_m, \quad \forall m \in M \setminus \{1\}; \quad (8)$$

$$\sum_{i \in C} (1 - s_{i,(m-1)}) \geq y_m, \quad \forall m \in M \setminus \{1\}; \quad (9)$$

Constraints (5) guarantee that the m -th visit will happen if and only if the previous visit occurs as well. Constraints (6) link the variable that determines if client i is served on visit m , p_{im} , to the variable that memorizes the clients that have already been served, s_{im} . For instance, if client 4 is served on visit number 3, $p_{4,3} = 1$ and, therefore, $s_{4,3} = 1$. Constraints (7) ensure that, if client i has already been served, then this information is passed on to future visits. Constraints (8) and (9) establish the drone's stopping point, that is, if all the clients have already been served on the m -th visit, then there are no more visits from the drone.

- Load constraints

$$q_j \leq q_i + l_j + K(1 - \sum_{m \in M} x_{ijm}), \quad \forall i \in V, j \in C, i \neq j; \quad (10)$$

$$q_j \geq q_i + l_j - K(1 - \sum_{m \in M} x_{ijm}), \quad \forall i \in V, j \in C, i \neq j; \quad (11)$$

$$l_i \leq q_i \leq \min\{Q_e, Q_d\}, \quad \forall i \in C; \quad (12)$$

$$q_0 = 0; \quad (13)$$

Constraints (10) and (11) determine the load that has already been delivered by the deliveryman after serving client j , preceded by client i , if the route goes from i to j . Constraints (12) guarantee that the load capacity will never be exceeded. Constraint (13) indicates the total cargo delivered at the start of every route.

- Time constraints

$$t_j \leq t_i + \frac{d_{ij}}{v_e} + K(1 - \sum_{m \in M} x_{ijm}), \quad \forall i \in C, j \in C, i \neq j; \quad (14)$$

$$t_j \geq t_i + \frac{d_{ij}}{v_e} - K(1 - \sum_{m \in M} x_{ijm}), \quad \forall i \in C, j \in C, i \neq j; \quad (15)$$

$$t_j \leq \gamma_m + \frac{d_{0j}}{v_e} + K(1 - x_{0jm}), \quad \forall j \in C, \forall m \in M; \quad (16)$$

$$t_j \geq \gamma_m + \frac{d_{0j}}{v_e} - K(1 - x_{0jm}), \quad \forall j \in C, \forall m \in M; \quad (17)$$

Constraints (14) and (15) refer to the delivery times, meaning the time when the deliveryman starts serving client j , if he serves client i before, is the time when he arrived on client i plus the time to travel between i and j . Constraints (16) and (17) refer to the times of the first deliveries of each route, in which the first client to be served in the m -th visit depends on the time the drone arrives at the drop-off point, and the time to travel from the drop-off point to client j .

- Time constraints for the last deliveries in each route

$$a_{im} \leq t_i + \frac{d_{i0}}{v_e} + K(1 - x_{i0m}), \quad \forall i \in C, \forall m \in M; \quad (18)$$

$$a_{im} \geq t_i + \frac{d_{i0}}{v_e} - K(1 - x_{i0m}), \quad \forall i \in C, \forall m \in M; \quad (19)$$

$$a_{im} \leq Kx_{i0m}, \quad \forall i \in C, \forall m \in M; \quad (20)$$

$$T \geq a_{im}, \quad \forall i \in C, \forall m \in M; \quad (21)$$

$$T \leq T_{max}; \quad (22)$$

Constraints (18) and (19) determine the time when the deliveryman returns to the drone's drop-off point in each visit m , similarly to constraints (14) and (15). Constraints (20) guarantee that the time a_{im} will be zero if i is not the last client to be served in the m -th visit. Constraints (21) define the floor for the time of last delivery T . Constraint (22) limits the time for the last delivery to the maximum delivery time.

- The drone's time constraints

$$\gamma_m \geq 2\left(\tau + \frac{h}{v_d}\right) + \gamma_{m-1} - K(1 - y_m), \quad \forall m \in M \setminus \{1\}; \quad (23)$$

$$\gamma_1 = \tau + \frac{h}{v_d}; \quad (24)$$

Constraints (23) determine the minimum intervals between each of the drone's visits, which are the time to travel back and forth between the depot and the drop-off point if the m -th visit happens. Constraint (24) establishes the time of the first visit.

- Constraints synchronizing the drone and the deliveryman

$$a_{im} \geq \gamma_m - K(1 - x_{i0m}), \quad \forall i \in C, \forall m \in M; \quad (25)$$

$$a_{i(m-1)} \leq \gamma_m + K(1 - x_{i0(m-1)}) \quad \forall i \in C, \forall m \in M \setminus \{1\}; \quad (26)$$

Constraints (25) ensure that the deliveryman's m -th route ends after the drone's m -th visit, if visit m happens. Constraints (26) guarantee that the deliveryman will be at the drone's drop-off point when the drone arrives.

$$x_{ijm} \in \{0, 1\}, \quad \forall i \in V, \forall j \in V, \forall m \in M; \quad (27)$$

$$p_{im} \in \{0, 1\}, \quad \forall i \in V, \forall m \in M; \quad (28)$$

$$s_{im} \in \{0, 1\}, \quad \forall i \in V, \forall m \in M; \quad (29)$$

$$y_m \in \{0, 1\}, \quad \forall m \in M; \quad (30)$$

$$t_i \geq 0, \quad \forall i \in C; \quad (31)$$

$$a_{im} \geq 0, \quad \forall i \in C, \forall m \in M; \quad (32)$$

$$\gamma_m \geq 0, \quad \forall m \in M. \quad (33)$$

The domain of the variables is described in (27) - (33)

- Objective function

$$\text{Minimize } T \quad (34)$$

The objective function (34) aims to minimize the total operational time T . Thus, the model for the DATSP can be written as such:

$$\begin{aligned} &\text{Minimize } T \\ &\text{subject to: } (1) - (33) \end{aligned}$$

4. Computational experiments

In this section the computational experiments and their results are presented.

4.1. Test environment

The model was implemented using the programming language Python v.2.7.3 and the results were obtained using the optimization solver GUROBI v.7.0.1. The tests were executed in a computer with an Intel(R) Xeon(R) CPU E5-2620 0 @ 2.00GHz processor and 70 GB of RAM in a Linux Ubuntu v.3.5.0 x64 operating system.

4.2. Test instances data

Five test instances were developed for the problem, containing 4, 9, 16, 25 and 32 customers (see Table 2), randomly distributed on a $10km$ by $10km$ field, where the depot is located at the bottom-left corner. The demands of these clients were generated randomly between 1, 2, and 3 units of weight, and the load capacities were of 4 units of weight for the drone and 5 units of weight for the deliveryman. The average speeds were defined as $60km/h$ for the drone and $40km/h$ for the deliveryman. The distance between the depot and the drone's drop-off point is $2km$, while the time for the drone to take off and land was defined as 1 minute. Moreover, M was defined as a set $\{1, \dots, n\}$, in which n is the number of clients in a given instance, since n is the maximum number of visits necessary to serve every client. The maximum operational time was defined as $T_{max} = 30h$. The values considered for K in each constraint are described in Table 1.

Table 1: Calculation of the value given to K in each constraint.

No.	Constraint	Value
(10)	$q_j \leq q_i + l_j + K(1 - \sum_{m \in M} x_{ijm})$	$K = \min\{Qe, Qd\}$
(11)	$q_j \geq q_i + l_j - K(1 - \sum_{m \in M} x_{ijm})$	$K = \min\{Qe, Qd\} + \max_{i \in C}\{l_i\}$
(14)	$t_j \leq t_i + \frac{d_{ij}}{v_e} + K(1 - \sum_{m \in M} x_{ijm})$	$K = T_{max}$
(15)	$t_j \geq t_i + \frac{d_{ij}}{v_e} - K(1 - \sum_{m \in M} x_{ijm})$	$K = T_{max} + \max_{i,j \in C}\{\frac{d_{i,j}}{v_e}\}$
(16)	$t_j \leq \gamma_m + \frac{d_{0j}}{v_e} + K(1 - x_{0jm})$	$K = T_{max}$
(17)	$t_j \geq \gamma_m + \frac{d_{0j}}{v_e} - K(1 - x_{0jm})$	$K = T_{max} + \max_{j \in C}\{\frac{d_{0,j}}{v_e}\}$
(18)	$a_{im} \leq t_i + \frac{d_{i0}}{v_e} + K(1 - x_{i0m})$	$K = T_{max}$
(19)	$a_{im} \geq t_i + \frac{d_{i0}}{v_e} - K(1 - x_{i0m})$	$K = T_{max} + \max_{i \in C}\{\frac{d_{i,0}}{v_e}\}$
(20)	$a_{im} \leq Kx_{i0m}$	$K = T_{max}$
(23)	$\gamma_m \geq 2(\tau + \frac{h}{v_d}) + \gamma_{m-1} - K(1 - y_m)$	$K = 2(\tau + \frac{h}{v_d}) + T_{max}$
(25)	$a_{im} \geq \gamma_m - K(1 - x_{i0m})$	$K = T_{max}$
(26)	$a_{i,m-1} \leq \gamma_m + K(1 - x_{i0(m-1)})$	$K = T_{max}$

4.3. Results

Table 2 presents the results obtained using the model and test instances. The maximum execution time was defined as 3600 seconds. The first two columns of the table indicate the instance number and the number of clients. The solutions for each instance are in column *O.F.* (Objective Function). The fourth column, *Bound*, presents the values of the lower bounds. Column *Visits*

shows how many visits the drone made to the drop-off point in each instance. In *Gap* the relative gap values are presented. Column *Execution time* displays the execution times in seconds. The last two columns present the number of variables and constraints generated in each instance.

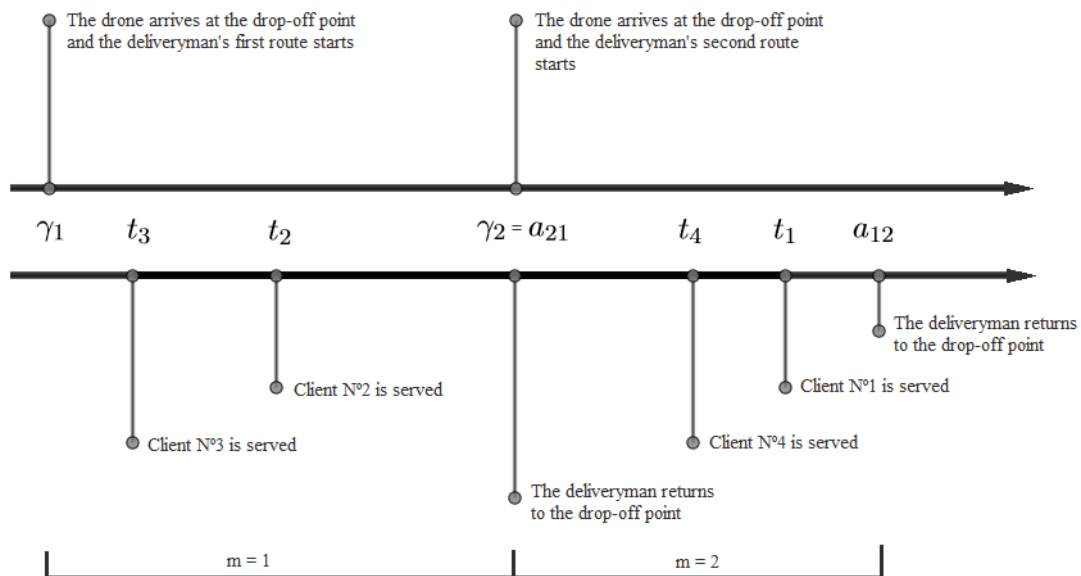
Table 2: Results for the instances obtained using the solver GUROBI.

Inst.	Clients	Model					Dimensions	
		O.F. (h)	Bound	Visits	Gap	Execution time (s)	Variables	Constraints
1	4	1.04	1.04	2	0.00	< 1	146	297
2	8	2.01	2.01	4	0.00	77	802	1142
3	15	3.63	0.70	9	0.81	3600	4337	3928
4	24	6.11	0.30	13	0.95	3600	16226	9958
5	32	7.56	0.06	17	0.99	3600	36994	17630

Analyzing the results obtained, it is clear that the dimensions of the problem grow immensely when the number of clients is increased. It is also worth noting that, for smaller instances, the solver found the optimal solutions quickly, but cannot do so for larger instances. One aspect that might be responsible for the poor quality of the dual bounds of the problem is the amount of constraints that include the parameter K .

Figure 3 displays the order of events in the solution for instance 1. Firstly, the drone arrives at the drop-off point (γ_1 , that is, $m = 1$), bringing with it the products for customers 2 and 3. The deliveryman then travels a route defined by the TSP between the two, going first to client 3 (time t_3), and then to client 2 (time t_2), returning to the drop-off point at time a_{21} . The drone's second visit (γ_2 , $m = 2$) occurs at the same time the deliveryman returns from his first route. The drone does not wait at the drop-off point, as expected. The deliveryman then serves the remaining customers, 4 (time t_4) and 1 (time t_1), returning to the drop-off point at time a_{12} . As there are no more clients left, the drone does not make any new trips, marking the end of the operation.

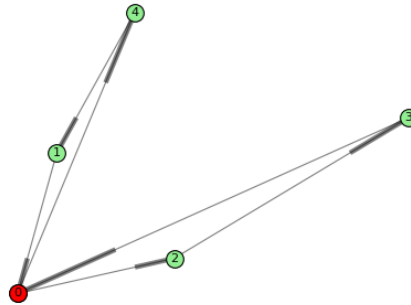
Figure 3: Timeline of events for Instance 1.



Source: Authors.

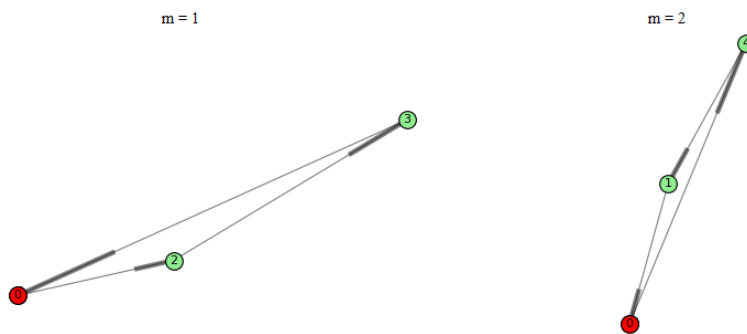
Figure 4 shows the complete route for instance 1 in the same order as presented in Figure 3. Figure 5 illustrates the routes taken by the deliveryman during each of the drone's visits.

Figure 4: The complete route traveled by the deliveryman.



Source: Authors.

Figure 5: Individual routes taken by the deliveryman in each visit.



Source: Authors.

5. Concluding Remarks

In this article a preliminary study on the Drone-Assisted Traveling Salesman Problem (DATSP) was introduced. The DATSP is a new type of problem wherein a drone flies over an obstacle carrying products to a deliveryman. The authors establish that there is commercial relevance in this type of application, given that several companies have shown interest in using drones. A mathematical model elaborated as a Mixed Integer Linear Programming (MILP) problem was developed so as to represent the DATSP.

In order to test the model, five different instances were created and then solved using the solver GUROBI. The results showed that the mathematical model behaves as expected, providing optimal solutions for small instances. However, the optimal solution is not found for larger instances within the time limit, due both to the poor quality of the dual bounds and to the problem's NP-Hard nature, since the complexity of the problem grows exponentially as the number of clients increases.

Since the model does not provide optimal solutions for instances of medium and large scales, the study of heuristics or other solution methods is the starting point for any future research. Investigations concerning the mathematical model can be made so as to bring it closer to reality, as

the current model does not consider the drone's battery consumption, which may change depending on the weight being carried, the total flight time, and even the drone's model. Another perspective could be to expand the current model for multiple drones or multiple deliverymen.

Acknowledgements

The authors would like to thank CNPq, CAPES, and CeMEAI-CEPID (FAPESP No. 2013/07375-0) for the financial support received.

References

- CBS (2013). Amazon unveils futuristic plan: Delivery by drone. URL <https://www.cbsnews.com/news/amazon-unveils-futuristic-plan-delivery-by-drone/>.
- DHL (2016). Successful trial integration of dhl parcelcopter into logistics chain. URL http://www.dhl.com/en/press/releases/releases_2016/all/parcel_ecommerce/successful_trial_integration_dhl_parcelcopter_logistics_chain.html.
- Dorling, K., Heinrichs, J., Messier, G. G., and Magierowski, S. (2017). Vehicle routing problems for drone delivery. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1):70–85.
- Gilchrist, K. (2017). World's first drone delivery service launches in iceland. URL <https://www.cnbc.com/2017/08/22/worlds-first-drone-delivery-service-launches-in-iceland.html>.
- Lithia, F. (2017). Ups tests residential delivery via drone launched from atop package car. URL <https://www.pressroom.ups.com/pressroom/ContentDetailsViewer.page?ConceptType=PressReleases&id=1487687844847-162>.
- Murray, C. C. and Chu, A. G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54: 86–109. URL <https://doi.org/10.1016/j.trc.2015.03.005>.
- Rose, C. (2015). Amazons jeff bezos looks to the future. URL <https://www.cbsnews.com/news/amazons-jeff-bezos-looks-to-the-future/>.
- Scholz, N. (2016). Pizza-by-drone a reality with world-first customer deliveries in new zealand. URL <https://www.dominos.com.au/inside-dominos/media/november-2016-pizza-by-drone-a-reality-with-world-first-customer-deliveries-in-new-zealand>.
- Simmons, D. (2016). Rwanda begins zipline commercial drone deliveries. URL <http://www.bbc.com/news/technology-37646474>.
- Stolaroff, J. and Samaras, C. (2018). Delivering packages with drones may benefit the environment. URL <https://www.weforum.org/agenda/2018/02/delivering-packages-with-drones-might-be-good-for-the-environment>.
- Stolaroff, J. K., Samaras, C., O'Neill, E. R., Lubers, A., Mitchell, A. S., and Ceperley, D. (2018). Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nature Communications*, 9(1). URL <https://doi.org/10.1038/s41467-017-02411-5>.