RAIRO-Oper. Res. 56 (2022) 853–869
https://doi.org/10.1051/ro/2022036

RAIRO Operations Research
www.rairo-ro.org

UAVS ROUTES OPTIMIZATION ON SMART CITIES AND REGIONS

Elias L. Marques Jr.[1,*], Vitor N. Coelho[2], Igor M. Coelho[1],
Yuri A. de M. Frota[1], Roozbeh H. Koochaksaraei[3],
Luiz Satoru Ochi[1] and Bruno N. Coelho[4]

Abstract. Unmanned Aerial Vehicles are becoming a common technology used on smart cities and smart regions, thus requiring optimization of its routes with crucial importance. In this innovative work, six objective functions are optimized in order to provide sets of non-dominated solutions, composed of routes with different characteristics. Realistic constraints are considered such as obstacles and areas in which drones could not pass through. A didactic case of study considering points of a graph is used in order to illustrate a smart city composed of different regions. Obtained solutions are analyzed using a state-of-the-art visualization tool, which guides the comprehension of harmony and conflicts between objectives.

Mathematics Subject Classification. 90C29, 05C85.

Received November 13, 2021. Accepted February 28, 2022.

1. Introduction

Recent technological advances have pushed towards the development and practical adoption of novel aerial transportation methods, under the topic of Unmanned Aerial Vehicles (UAV).

Although UAV is often related to the hobby, entertainment and photographic industries, its uses have spread to military, civil and commercial applications. Aerial surveillance, object recognition and tracking are some of the many other applications that are emerging with potential for the use of UV (unmanned vehicle). Countless others may arise from human creativity in the near future [6].

The freight transport sector, in particular, is already showing some interest and investment in UV applications. The growth of e-commerce supported this interest from large companies. Transport drones are able to safely take off and land near buildings and humans, improving the quality of service today in congested or remote areas [10].

Keywords. Smart City, UAVs, routing problem, MILP, many-objective optimization problem.

[1] Institute of Computing, Universidade Federal Fluminense, Av. Gal. Milton Tavares de Souza, São Domingos, Niterói, RJ 24210-310, Brazil.

[2] OptBlocks Consultoria Ltda., Avenida João Pinheiro, 274 Sala 201, Lourdes, Belo Horizonte, MG 30130-186, Brazil.

[3] Adjunct Faculty, Computer Science Department of George Washington University, Washington, DC 20052, USA.

[4] Department of Control and Automation Engineering, Universidade Federal de Ouro Preto, Campus Morro do Cruzeiro, Ouro Preto, MG 35400-000, Brazil.

[*]Corresponding author: eliaslawrence@id.uff.br

Typical limitations of UAV include: limited energy storage; limited payload weight; limited speed (according to weight); limited flight range (according to flight management and regulation). On the other hand, UAV allows accessing remote locations and greater degrees of freedom during movement, when compared to terrestrial vehicles. This calls for complex optimization processes to emerge when routing UAV through a set of client points.

When the delivery service is discussed, we are implicitly talking about a Traveling Salesman Problem (TSP) and its variations, like the Vehicle Routing Problem (VRP), for example. Which briefly means a problem of designing optimal routes from one or several depots to a number of geographically dispersed cities, customers or strategic points, subject to lateral constraints [20]. Although there are many works in the literature related to variations of the TSP [11], those that approach UV routing are still few.

However, we cannot just focus on technological advances and simply forget about the damage to the environment they can cause. That's why the scientific community has been so concerned about developing green technologies and it is no different in computing. The Green Vehicle Routing Problem (G-VRP), for example, proposed by Erdoan *et al.* [9], adds to the original VRP, restrictions on fuel economy and/or choosing the least environmentally harmful fuel.

The TSP with hotel selection [30] is a variation of the TSP with similarities to the problem addressed in this work. The main objective is to minimize the number of trips and total travel time. This problem is found in real-world scenarios, such as the delivery of products by electric vehicles that need to be recharged during a ride.

Complex, real-world systems are usually composed by dozens or hundreds of variables and constraints. However, covering 100% of it is usually not feasible, it is often necessary to design a multi-objective problem providing a set of non-dominated solutions (pool of solutions) with different possible routes and times. In UAV problems, for example, it is likeable to find a solution that minimizes the route time and consumption at the same time. The more objective functions and constraints it contains, the model tends to be more similar to the real world and more complex. So, a trade-off must be done.

The main contributions of this current work are:

- Improve mathematical models from the literature and introduce a novel programming model for a time-dependent UV routing problem, considering the following restrictions:
  - respecting the operational requirements of the UVs;
  - addressing micro-airspace avoiding prohibited points (docking constraints) [5];
  - integrating UVs into new concepts of microgrid systems, in which vehicles can be loaded and/or recharged at different points in future smart cities;
  - dynamic routes considering drones already in motion: instances with initial battery different from 100% and random point of origin;
  - Presents a realistic scenario obtained from real-world from huge cities located in Brazil.

The remainder of this paper is organized as follows. Section 2 enlists works related to the theme of this paper. Section 3 describes the proposed model of the Multi-Objective Green Routing Drone Grid Problem (MOGRDGP), while Section 4 contains the mathematical formulation to tackle the problem. Section shows the instances created based on Brazilian regions and geographical sites as constraints. In Section 6, one can find the computational experiments comparing the different implementations, instances, variables and results. Finally, Section 7 concludes the work and presents future research directions.

2. Related work

The work of Coelho *et al.* [6] proposed a mathematical programming model for the multi-objective UAV routing problem, seeking to minimize seven distinct strategical objective functions while respecting UAV battery constraints. On the other hand, a crucial set of constraints that involve avoiding prohibited points was not considered by the latter. Here, the linear mathematical model is complemented in order to tackle the micro-airspace considering docking constraints [5] for inspection points.

Table 1. Comparison between UAV routing problem publications.

	This paper	Coelho *et al.* [6]	TSPD	k-MVDRP [27]	FSTSP [17]
Microgrid	✓	✓			
Multiobjective	✓	✓			
Docking constraint	✓				✓
No. of Drones	Multiple	Multiple	Single	Multiple	Single
Multilayer		✓			
Deliver/collect		✓		✓	✓
Truck			✓	✓	✓

The current problem stems from the Traveling Salesman Problem with Drones (TSPD) [2] and the Vehicle Routing Problem with Drones (VRPD) [31]. Both problems address the visit of clients by drones, used as complementary vehicles to a main one (such as a truck). On the contrary, TSPD and VRPD are represented by graphs, which differs from the grid representation used in this work.

In addition, the current model integrates UAVs into the new concepts of microgrid systems, in which vehicles can be charged at different points of smart cities/regions, as well as consider dynamic in-route drones (which allows the system to be re-optimized at any time, even when a given solution is already being implemented). The latter allows new clients to be added or removed at any time, as well as other changes on environment conditions. It is based on the following principles: allows initial battery to be different than 100% and permits random origin point.

The scope of the model introduced in this paper is not limited to deliver activities, differently than Coelho *et al.* [6]. In this sense, it takes into account the distinct lack of applications that UAVs can cover such as infrastructure inspection [1, 8, 16, 22, 24], surveillance [14, 25], area mapping [12] and others. Focusing, then, on realistic assumptions required for fast re-optimization (*via* heuristics) and taking into account many requirements like restricted area.

Moreover, this study tries to analyze the many objective problem space to make sure that the optimization algorithms perform with high performance output. One of the modern approaches to have better problem space understanding is throughout visualization techniques. The applied data visualization tool provides a qualitative analyzing with Schneiderman's mantra [29] to present two level of details [13] of the problem space. This strategy has been applied in a multidisciplinary form of many objective optimization [3, 18, 23]. This approach enables subject matter experts to understand the relationships between objectives in a problem space and then, based on the amount of harmonies and conflicts and also according to the priorities, use dimension reduction methods on the problem space. It is crystal clear that reduction of objectives can increase the chance of finding global optimum solutions in problem space.

Table 1 compares some prominent works in the UAV routing field with our current work. The main differentials of this paper to the literature, as we can observe, are the microgrid approach allied to the docking constraint. Drone routing problems generally assume a deterministic distance between two nodes and the grid modelling tackles this issue.

Incorporating spatial constraints (*e.g.*, no fly zones) is also an important issue to create more realistic models. Certain areas (*e.g.*, airports, government or critical facilities, areas subject to communication interference or interruptions) will likely remain off-limits to drone fly-over, so optimal routes need to incorporate those restrictions [15, 26].

3. Problem description

The proposed MOGRDGP consists of an airspace divided into horizontal and vertical bands, organized as a grid of points in two-dimensional space. Each UAV can move following the Chebyshev distance, where the

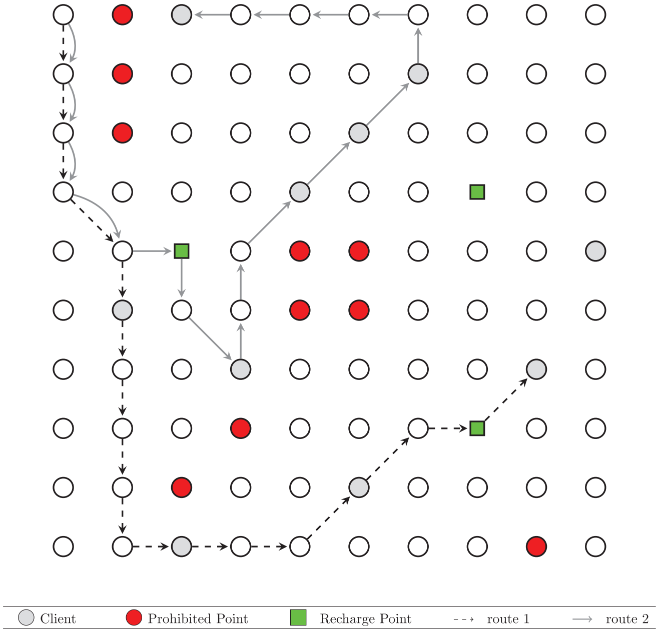


Figure 1. Instance and solution example to MOGRDGP.

distances between any adjacent points are the same. This distance metric was chosen to simplify the calculations, without loss of generality, and without prejudice to the construction of the routes.

Power stations are scattered around the routing area and accessed by the drone to recharge its batteries. To represent prohibited areas, the grid is also composed of prohibited points that the UAV cannot access, otherwise, this invalidates the route. Figure 1 illustrates a solution for a PMORVD instance.

As a routing problem, the vehicle must serve/visit clients that are spread across the grid. For this, the point corresponding to the client must be part of the final route. This means that the clients' x and y coordinates must be part of the matrix that represents the solution.

In order to model the problem computationally, we mapped five main objectives that satisfactorily summarize how the real system should behave. It is desirable to end the route with the maximum possible charge rate (*final charge* – O4), ensuring that the drone is prepared for a future route.

This means that a drone can start a new route at the end of an old one, as well as in the middle of the route if the conditions change. The total route must be completed in the shortest possible time, represented by the combination of maximum lowest speed (O0 – maximizes the lower speed limit, then maximizing the UAV speed throughout the whole route), total travelled distance (O1) and time spent recharging the vehicle (O2). Also, it is desirable that during the route, the vehicle consumes (*consumption* – O3) as little battery/fuel as possible.

If we compare the objective functions of this work with the ones explored in [6], it can be related O4 with *toFull*, O1 with *distance*, O0 with *maxvel* and O3 is combination of *time*, *distance* and *maxvel* in a certain way. The number of drones was not seen as an objective function that should be minimized but as a parameter of the instances to test the developed methods. Since our focus was to elaborate a model that could fit to different applications involving drones, constraints and objectives exclusively related to delivery problems; addressed in [6] such as *makespanC*, *makespanD*, weight of the products, capacity of the UAVs; were not taken into consideration in this work.

The algorithm proposed in this article focuses on finding a balance between solutions, since one element can affect another. The shorter the time, the greater the speed, the greater the consumption. Higher final charge means more time spent on recharging/refueling, which results in longer times.

4. MILP

In this section, we propose the mathematical formulation of MOGRDGP.

4.1. Sets, parameters, auxiliary and decision variables

The following parameters were defined and considered for the model:

Parameters	
T	Set of discretized time intervals $\{1, \cdots, t_{\max}\}$
$TimeUnit$	Fixed time unit at any time between $t \in T$ and $t+1 \in T$
Z_x	Length of the map to be considered by the model
Z_y	Width of the map to be considered by the model
C	Set of clients, containing the following information for each client c
$C_{x,c}$	x coordinate for $c \in C$
$C_{y,c}$	y coordinate for $c \in C$
E	Set of energy stations
$E_{x,e}$	x coordinate for $e \in E$
$E_{y,e}$	y coordinate for $e \in E$
P	Set of map points that the vehicle cannot pass through.
$P_{x,p}$	x coordinate for $p \in P$
$P_{y,p}$	y coordinate for $p \in P$
U	Set of available UAVs of predetermined size in the instance
I_x	x coordinate of UAVs at the beginning of the route ($t = 1$)
I_y	y coordinate of UAVs at the beginning of the route ($t = 1$)
U_{count}	Number of UAVs to be used in the solution
$V_{\max}$	Maximum speed of UAVs

Vehicle recharge parameters:

Vehicle recharge parameters	
DOR	Duration of the recharge process per full charge
VEV	Variable energy consumption (%) related to UAV speed
FEV	Fixed consumption (%) at any time between $t \in T$ and $t+1 \in T$
BI	Percentage of drone battery at start (%)

The following decision variables used to provide a complete solution for the proposed model.

Decision variables	
$vC_{u,c,t}$	Binary variable that indicates whether the client c was visited by UAV u at time t
$vE_{u,e,t}$	Binary variable that indicates whether the station e was visited by UAV u at time t
$pos^x_{u,t}$	Coordinate $x \in \mathbb{Z}_+$ of UAV u in time t
$pos^y_{u,t}$	Coordinate $y \in \mathbb{Z}_+$ of UAV u in time t
$v_{u,t}$	Speed of UAV u in time t (starting in the interval $t-1$ to $t \in T$)
$on_{u,t}$	UAV u is in operation at time t, binary

About the prohibited area constraints, the following decision variables were used. Since the model of the problem was based on a grid, four variables were needed, each one representing a quadrant of the Cartesian plane.

No-fly zone decision variables	
$uP^{\ngeq,x}_{u,p,t}$	Binary variable that indicates whether the UAV u coordinate x is not greater than the coordinate x of the forbidden point p at time t
$uP^{\nleq,x}_{u,p,t}$	Binary variable that indicates whether the UAV u coordinate x is not less than the coordinate x of the forbidden point p at time t
$uP^{\ngeq,y}_{u,p,t}$	Binary variable that indicates whether the UAV u coordinate y is not greater than the coordinate y of the forbidden point p at time t
$uP^{\nleq,y}_{u,p,t}$	Binary variable that indicates whether the UAV u coordinate y is not less than the coordinate y of the forbidden point p at time t

Furthermore, additional auxiliary variables were needed to update the battery rate of the UAV at each t interval, as well as checking whether the UAVs are running at each t time.

$rechargeRate_{u,e,t}$	Recharge rate of UAV u at station e at time t (%)
$batRate_{u,t}$	Battery charge of UAV u in time t, ≥ 0 and ≤ 100

4.2. Goals to be optimized

The route must be carried out in the shortest possible time, thus we have to minimize the total travelled *distance* (Obj1), maximize the *minVel* (Obj0) of the vehicles and reduce the time spent at base stations (*rechargeTime* – Obj2). The equations (4.1), (4.2) and (4.3) represent these goals respectively.

$$distance \geq \sum_{t \in T} on_{u,t} \qquad \forall u \in U \tag{4.1}$$

$$minVel \leq v_{u,t} \qquad \forall u \in U, t \in T : t \leq t_{\max} \tag{4.2}$$

$$rechargeTime \geq \sum_{e \in E} \sum_{t \in T} rechargeRate_{u,e,t} \qquad \forall u \in U. \tag{4.3}$$

cons (Obj3): it is desirable that during the route, the vehicle consumes as little battery/fuel as possible. Equation (4.4) measures the consumption of all vehicles. The clarification of the variables is presented in the Section 4.4.

$$cons \geq \sum_{t \in T} \mathrm{VEV} \cdot \frac{v_{u,t}}{V_{\max}} + on_{u,t} \cdot \mathrm{FEV} \qquad \forall u \in U. \tag{4.4}$$

The *cons* variable is composed by two parts, the first one responsible for representing the velocity dependent element (greater the speed on the stretch, greater the variable consumption) and the other one representing the fixed portion.

finalCharge (Obj4): it is interesting to end the route with the maximum possible charge rate, ensuring that the drone is prepared for a future route, since the current work has a dynamic nature where the instances already consider the initial location and capacity of the drone to be arbitrary, which means that a drone can start a new route at the end of an old one, as well as in an arbitrary point. Equation (4.5) measures the minimum amount of energy that UAVs will have at the end of the route.

$$finalCharge \leq batRate_{u,t_{\max}} \qquad \forall u \in U. \tag{4.5}$$

We then have the structure of the objective function demonstrated by equation (4.6). The λ_i parameters indicate the weights for each objective. Their values are set according to an algorithm explained on Section 4.5.

$$f.o. = \min\left(-\lambda_0 \cdot minVel + \lambda_1 \cdot distance + \lambda_2 \cdot rechargeTime + \lambda_3 \cdot cons - \lambda_4 \cdot \frac{finalCharge}{100}\right). \tag{4.6}$$

4.3. Constraints and operational requirements of the model

The initial position of the UAVs is updated in the equations (4.7) and (4.8).

$$pos^{x}_{u,1} = I_x \qquad \forall u \in U \tag{4.7}$$
$$pos^{y}_{u,1} = I_y \qquad \forall u \in U. \tag{4.8}$$

Equations (4.9) and (4.10) ensure that drones can only move to points adjacent to each shift.

$$-on_{u,t} \leq pos^{x}_{u,t} - pos^{x}_{u,t-1} \leq on_{u,t} \qquad \forall u \in U, t \in T : t \geq 2 \tag{4.9}$$
$$-on_{u,t} \leq pos^{y}_{u,t} - pos^{y}_{u,t-1} \leq on_{u,t} \qquad \forall u \in U, t \in T : t \geq 2. \tag{4.10}$$

Equations (4.11) and (4.12) refer to the map boundaries.

$$0 \leq pos^{y}_{u,t} \leq Z_y \qquad \forall u \in U, t \in T \tag{4.11}$$
$$0 \leq pos^{x}_{u,t} \leq Z_x \qquad \forall u \in U, t \in T. \tag{4.12}$$

From equation (4.13) to equation (4.16), if a client is visited, a UAV passes through its coordinates. The combination of these four equations represent the quadrants of the grid in wich the variable $vC_{u,c,t}$ can only be set to 1 (UAV u visited client c at time t) if both coordinates x and y of the UAV and client match.

$$pos^{x}_{u,t} - C_{x,c} \leq Z_x(1 - vC_{u,c,t}) \qquad \forall u \in U, c \in C, t \in T \tag{4.13}$$
$$-pos^{x}_{u,t} + C_{x,c} \leq Z_x(1 - vC_{u,c,t}) \qquad \forall u \in U, c \in C, t \in T \tag{4.14}$$
$$pos^{y}_{u,t} - C_{y,c} \leq Z_y(1 - vC_{u,c,t}) \qquad \forall u \in U, c \in C, t \in T \tag{4.15}$$
$$-pos^{y}_{u,t} + C_{y,c} \leq Z_y(1 - vC_{u,c,t}) \qquad \forall u \in U, c \in C, t \in T. \tag{4.16}$$

All clients must be visited by at least one UAV:

$$\sum_{u \in U} \sum_{t \in T} vC_{u,c,t} \geq 1 \qquad \forall c \in C. \tag{4.17}$$

Equations (4.18)–(4.22) concern the ban on the passage of drones at points reported as prohibited. If the route contains these points, the solution will be invalid. If every component of equation (4.22) is set to 1, it means that the UAV is passing through a no-fly area, which is a constraint of the problem.

$$pos^{x}_{u,t} - P_{x,p} \geq 1 - \left(Z_x \cdot uP^{\ngeq,x}_{u,p,t}\right) \qquad \forall u \in U, p \in P, t \in T \tag{4.18}$$
$$-\, pos^{x}_{u,t} + P_{x,p} \geq 1 - \left(Z_x \cdot uP^{\nleq,x}_{u,p,t}\right) \qquad \forall u \in U, p \in P, t \in T \tag{4.19}$$
$$pos^{y}_{u,t} - P_{y,p} \geq 1 - \left(Z_y \cdot uP^{\ngeq,y}_{u,p,t}\right) \qquad \forall u \in U, p \in P, t \in T \tag{4.20}$$
$$-\, pos^{y}_{u,t} + P_{y,p} \geq 1 - \left(Z_y \cdot uP^{\nleq,y}_{u,p,t}\right) \qquad \forall u \in U, p \in P, t \in T \tag{4.21}$$
$$uP^{\ngeq,x}_{u,p,t} + uP^{\nleq,x}_{u,p,t} + uP^{\ngeq,y}_{u,p,t} + uP^{\nleq,y}_{u,p,t} \leq 3 \qquad \forall u \in U, p \in P, t \in T. \tag{4.22}$$

Finally, equation (4.23) limits the speed of UAVs to the defined maximum.

$$on_{u,t} \leq v_{u,t} \leq V_{\max} \qquad \forall u \in U, t \in T. \tag{4.23}$$

4.4. Model restrictions and operational requirements for batteries

Restrictions related to UAV charging stations and the requirements for the charging rate value are listed below. Equation (4.24) defines the vehicle's initial battery charge, while equation (4.25) updates the battery rate each shift, increasing it if it has been recharged and decreasing it due to its speed and fixed consumption.

$$batRate_{u,1} = \text{BI} \qquad \forall u \in U. \tag{4.24}$$

The speed in the section influences not only the total time of the route, but also the consumption, the higher the speed (v), the greater the consumption. The fuel/battery level at the end of the stretch is a result of the fuel/battery level at the start of the stretch decreased by the fixed consumption (c_f) and the speed multiplied by the variable consumption coefficient as shown in equation (4.25).

The time spent at the power station also influences. If the vehicle spends more time, it can accumulate more fuel/energy for its battery. The fuel/battery level at the end of the stretch is a result of the fuel/battery level at the beginning of the stretch, increased by the amount of fuel/energy recharged.

$$batRate_{u,t} = batRate_{u,t-1} - \text{VEV} \cdot \frac{v_{u,t}}{V_{\max}} - on_{u,t} \cdot \text{FEV} + \sum_{e \in E} rechargeRate_{u,e,t} \qquad \forall u \in U, t \geq 2, t \in T \tag{4.25}$$

$$0 \leq batRate_{u,t} \leq 100 \qquad \forall u \in U, t \in T. \tag{4.26}$$

From equations (4.27) to (4.30) a UAV can recharge if it passes through the coordinates of the power stations. Here, these equations consist of the same idea of equations (4.13)–(4.16) applied to station visit instead of client one.

$$pos^x_{u,t} - E_{x,e} \leq Z_x(1 - vE_{u,e,t}) \qquad \forall u \in U, e \in E, t \in T \tag{4.27}$$

$$-pos^x_{u,t} + E_{x,e} \leq Z_x(1 - vE_{u,e,t}) \qquad \forall u \in U, e \in E, t \in T \tag{4.28}$$

$$pos^y_{u,t} - E_{y,e} \leq Z_y(1 - vE_{u,e,t}) \qquad \forall u \in U, e \in E, t \in T \tag{4.29}$$

$$-pos^y_{u,t} + E_{y,e} \leq Z_y(1 - vE_{u,e,t}) \qquad \forall u \in U, e \in E, t \in T. \tag{4.30}$$

The equation (4.31) represents the constraint in which the UAV can only charge at the station it visits.

$$\frac{rechargeRate_{u,e,t}}{100} \leq vE_{u,e,t} \qquad \forall u \in U, e \in E, t \in T. \tag{4.31}$$

As the variable $on_{u,t}$ informs the end of the route, the UAV can no longer recharge.

$$\frac{rechargeRate_{u,e,t}}{100} \leq on_{u,t} \qquad \forall u \in U, e \in E, t \in T. \tag{4.32}$$

4.5. Matheuristic

Considering the number of objectives of the proposed routing problem, it is advisable to obtain a set of non-dominated solutions. In order to do that in restricted computational time, it is proposed the use of the Multi-Objective Smart Pool Search (MOSPOOLS) Matheuristic, introduced by Coelho *et al.* [4] and extended in Coelho *et al.* [7].

A predefined set of objective function weights is defined as v$\Lambda = \{v\lambda_0, \ldots, v\lambda_i, \ldots, v\lambda_4\}$. Several different MILP problems are generated by the linear combination of the weights of each vector $v\lambda_i = \{v\lambda_i^1, \ldots, v\lambda_i^k\}$, composed of k possible weights. Thus, the Cartesian product of them, $\Lambda = \{v\lambda_0 \times \ldots \times v\lambda_4\}$, defines the number of MILP problems to be solved (Eq. (4.33)). This strategy is capable of providing a good balance between each objective generating a plural amount of solutions non-dominated.

$$|\Lambda| = |v\lambda_0| \times \ldots \times |v\lambda_i| \times \ldots \times |v\lambda_4|. \tag{4.33}$$

Algorithm 1. MOSPOOLS.

```
Input: Solver time limit $sTLimit$ and vector of MILP weights $\Lambda = \{v\lambda_0 \times \ldots \times v\lambda_4\}$
Output: Set of non-dominated solutions $Xe$
  $mipPop \leftarrow \emptyset$
  for all $\lambda_i \in \Lambda | \forall i = \{0, \ldots, 4\}$ do
      $model \leftarrow$ MILP model with weights $\lambda_i$
      $(poolSol, poolEval) \leftarrow MILPSolver(model, sTLimit)$
      for all $nS \in poolSol$ do
          $Xe \leftarrow addSolution(Xe, poolSol_{nS}, poolEval_{nS})$
      end for
  end for
  return $Xe$
```

Algorithm 1 states MOSPOOLS pseudocode, which calls the math model, described in Section 4, with a unique combination of weights for each of the seven objective functions. The generated model is solved and returns all feasible solutions obtained during the optimization process.

Each solution found is considered by the *addSolution* procedure (described in Algorithm 2). This procedure tries to add each solution from the pool of solution $s \in poolSol$ to the set of non-dominated solutions Xe.

Algorithm 2. addSolution.

```
Input: Population $Xe$ potentially efficient; Solution $s$, and its evaluations $z(s)$
Output: $Xe$
  for all $x \in Xe$ do
      if $z(x) \preceq z(s)$ then return $Xe$
      end if
      if $z(s) \prec z(x)$ then
          $Xe \leftarrow Xe \setminus x$
      end if
  end for
  $Xe \leftarrow Xe \cup s$
  return $Xe$
```

5. Case of study

The case of studies designed for this work are based on the airspace of one of the most active regions of Brazil (Fig. 2), and its main cities (Figs. 3–5), in terms of air and land traffic of people and merchandise.

As presented by Section 3, the airspace is represented by a grid of points of different characteristics (Figs. 6–8). The recharge points were assigned to the capitals, representing base stations in a real applications. The client points were assigned to other important cities surround the capitals. Some specific geographical sites, like the sea, river, environmental park were represented by prohibited points, since it would be advisable for UAVs not to pass by. These prohibited points could also represent any other forbidden area like military or private areas, big mountains or buildings, airports.

From the grid representing the case of study, it is possible to generate the MILP problem parameters C, P and E, respectively representing the sets of clients, prohibited points and recharge station. In order, the size of each set is 10, 14 and 3. The other parameters are listed in Table 2. The $t_{\max}$ was chosen as the total amount of points presented in the grid.

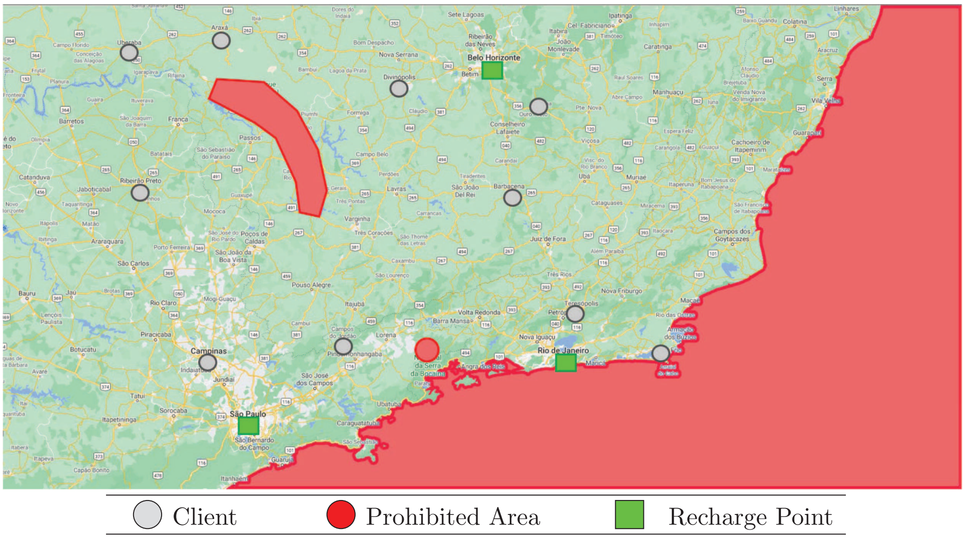


FIGURE 2. Brazil southeast (SE instance).

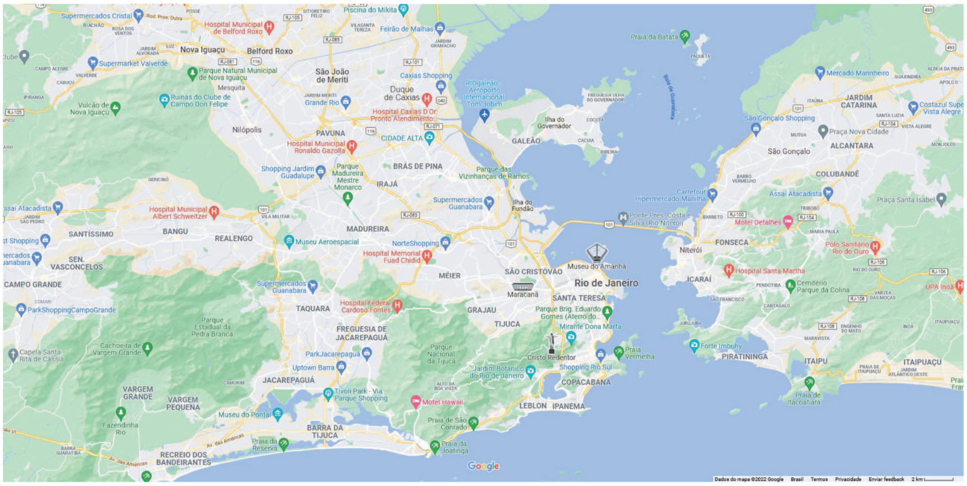


FIGURE 3. Rio de Janeiro (RJ) map representation.

6. COMPUTATIONAL EXPERIMENTS

Our experiments were performed with a virtual machine with 6 GB of virtual RAM and 4 vCPUs running a 64-bit version of Ubuntu 18.04 on the VirtualBox 5.0.10 hypervisor with Windows 10 as the host operating system. The host hardware configuration consists of an Intel Core i5-6400 CPU with 16 GB of RAM.

An implementation of the proposed MILP models can also be found at https://github.com/eliaslawrence/mogrdgp-mip. The model is implemented using Python-MIP [28]. To optimize the MILP model, the solver GUROBI 9.1.0 was used, a state-of-the-art commercial optimizer.

The following weights were adopted for each objective function (O_i), $\lambda_i = \{0.01, 0.1, 1\}$ in SE instance. For BH, RJ, and SP instance, it was adopted $\lambda_i = \{0.1, 1\}$.

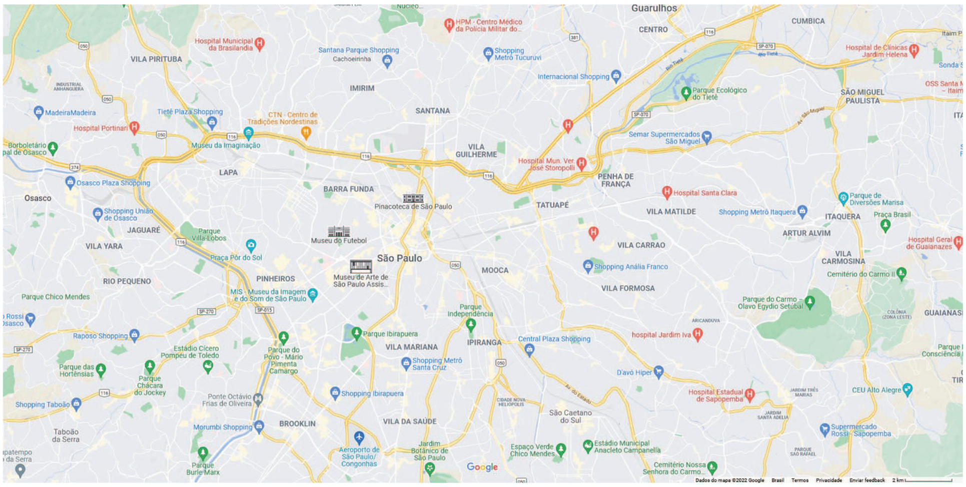


FIGURE 4. São Paulo (SP) map representation.

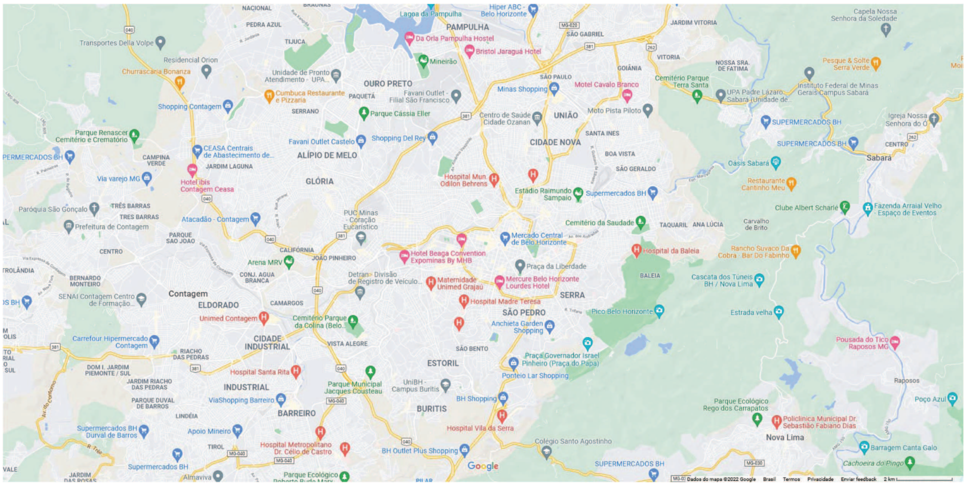


FIGURE 5. Belo Horizonte (BH) map representation.

Given all possible combinations of these sets of weights, 339 MILP problems (243 SE and 32 for each of the other instances) are going to be solved to find solutions for the cases of study. Five different time limits were established and the performance results can be seen in Tables 3 and 4.

For each execution we have the rate of the different configurations that found feasible solutions, optimal solutions and the average gap to the optimal solution. The last column present the number of non-dominated (ND) solutions, and some of the routes can be seen in Figures 9 and 10.

Multi-objective problems require a pool of non-dominated solutions as result. That being said, it can be found in the resulting set, solutions that favor a certain subset of objectives over the others depending on the combination of the sets of weights (λ_i), as can be seen in Tables 5 and 6. Than, the user must choose the solution that best fit the goals of the interest problem. For example, if we are talking about a rescue situation, the focus must be on "Lowest Speed" and "Distance"; problems of route in isolated area, the priority must be "Consumption".

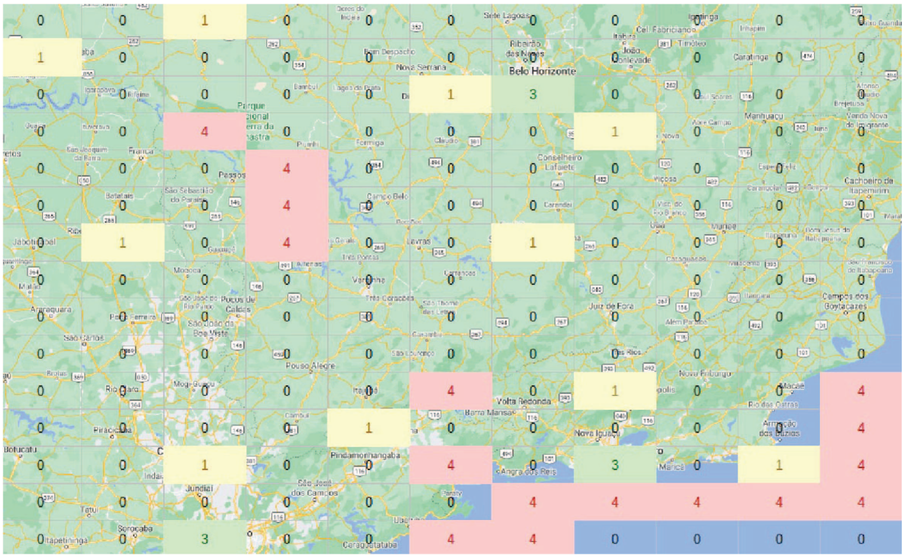
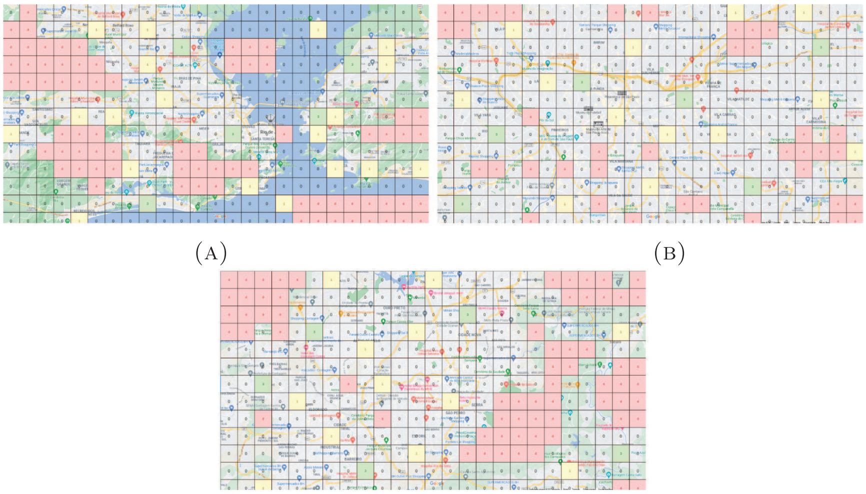


FIGURE 6. Converting region to grid (1 – client; 3 – station; 4 – prohibited area).



(A)

(B)

(C)

FIGURE 7. Converting cities to grid (1 – client; 3 – station; 4 – prohibited area). (a) Rio de Janeiro. (b) São Paulo. (c) Belo Horizonte.

Due to a high number of objective functions and high dimensionality requested for the trade-off visualization, non-dominated solutions are depicted using the tool proposed by Koochaksaraei *et al.* [19] (Fig. 11), a chord diagram based on parallel coordinates to analyze relationships between objectives. It can be seen that the lower boundary of Obj0 (Fig. 11a) is connected to the upper boundary of Obj1, Obj2 and Obj3. It means that the solutions with minimum value of Obj0 have high value in Obj1, Obj12 and Obj13.

Furthermore, with assistance of the chord diagrams (Fig. 11b), we concluded that distance (Obj1), rechargeTime (Obj2) and consumption (Obj3) could be minimized given the high harmony between these objectives,

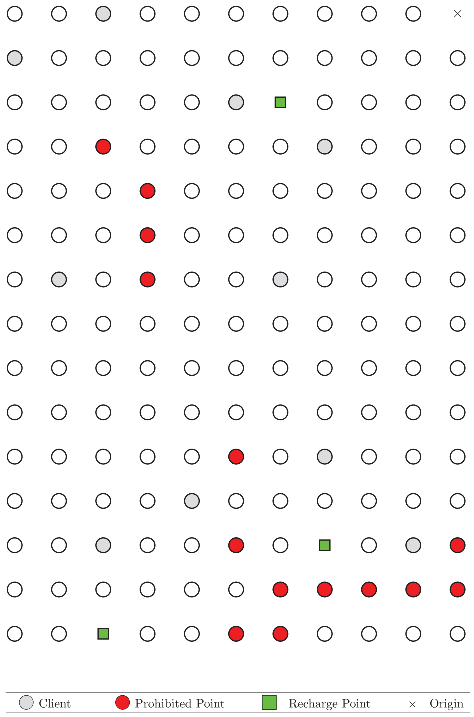


FIGURE 8. Grid representation of the southeast region.

TABLE 2. MILP parameters.

	Region	Cities
I_x	10	10
I_y	0	0
Z_x	10	12
Z_y	14	24
$t_{\max}$	$Z_x \cdot Z_y - 1$	$Z_x \cdot Z_y - 1$
U_{count}	1	1
$V_{\max}$	10	10
DOR	0.5	0.5
VEV	1	0.5
FEV	5	2
BI	100%	100%

both upper and lower boundaries are connected respectively. Obj0 and Obj4 do not present a direct correlation with each other, but we can also see a conflict of these objectives with Obj1, Obj2 and Obj3.

7. Conclusions

In this work, we approach the MOGRDGP, considering a novel and rich range of constraints, in addition to using a model with several objective functions. The problem was inspired by a global interest in using UAVs in several important applications. The multi-objective, grid, restrictions of prohibited areas (docking constraint),

TABLE 3. Performance results (SE instance).

MILP time limit	Feasible solutions	Optimal solutions	Average GAP	ND solutions
1 min	100%	0.00%	119.59%	19
5 min	100%	9.05%	68.02%	188
10 min	100%	21.81%	51.93%	174
15 min	100%	32.10%	40.69%	220
30 min	100%	53.91%	25.25%	244

TABLE 4. Cities maps performance results with 1 hour time constraint.

Map	Feasible solutions	Optimal solutions	Average GAP
RJ	100%	0.00%	90.91%
SP	100%	3.13%	65.90%
BH	100%	0.00%	87.11%

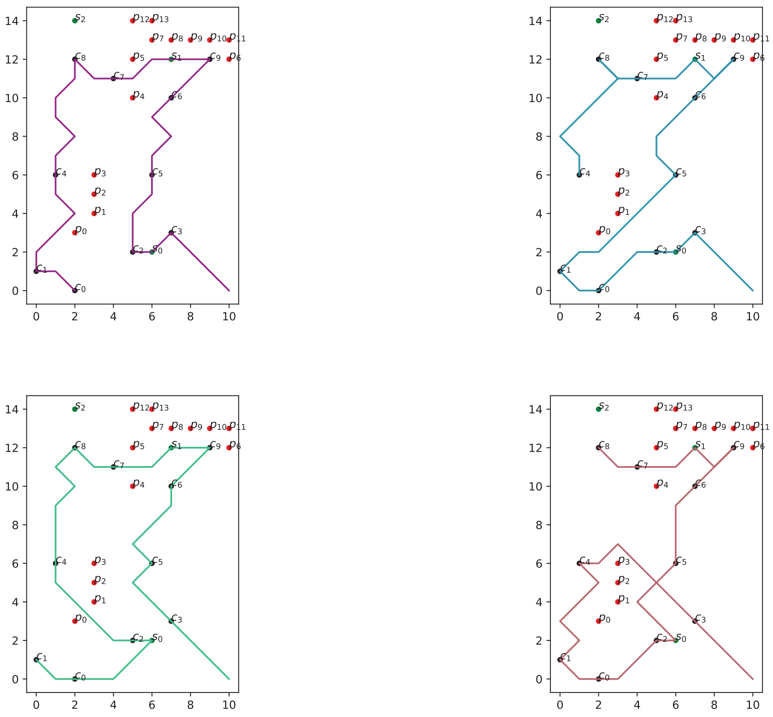
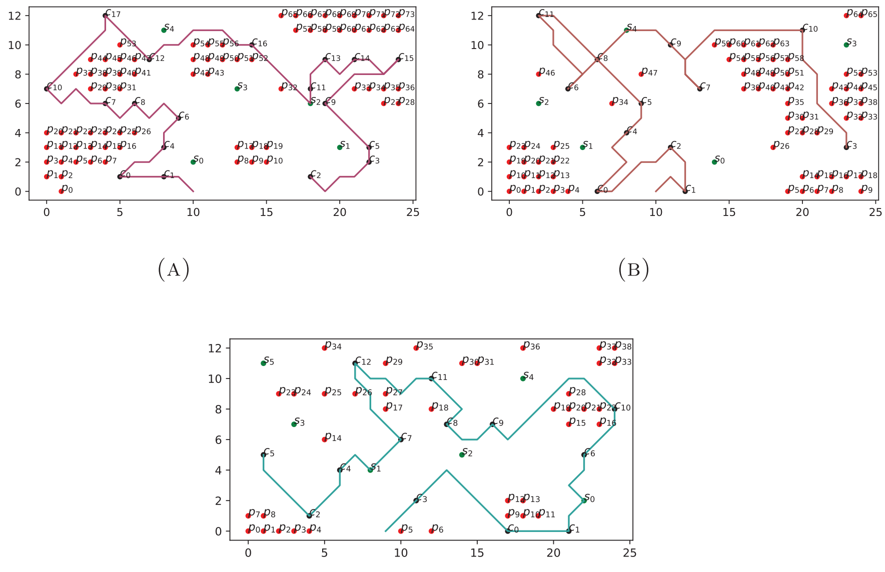


FIGURE 9. Graphic representation of some of the ND solutions found.



(A) (B)

(C)

Figure 10. Graphic representation of some of the ND solutions found. (a) RJ. (b) BH. (c) SP.

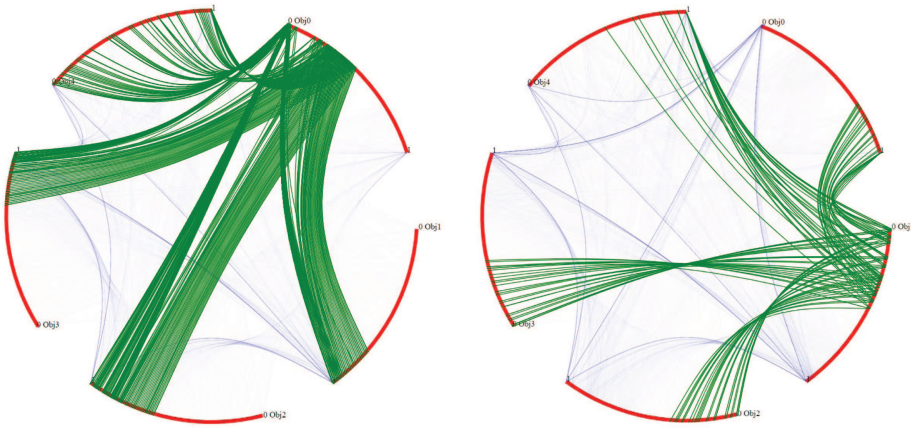
Table 5. Different non-dominated solutions characteristics (South-east region map).

Weights	Lowest speed	Distance	Recharge time	Consumption	Final charge rate (%)
$\{0.01, 1.00, 1.00, 0.10, 1.00\}$	1.0	36.0	1.2	183.6	41.0
$\{0.01, 0.01, 0.01, 1.00, 1.00\}$	10.0	42.0	2.22	252.0	76.0
$\{1.00, 0.01, 0.10, 1.00, 1.00\}$	2.0	42.0	2.14	219.2	100.0
$\{1.00, 0.01, 0.10, 0.10, 0.01\}$	10.0	41.0	2.4	246.0	100.0
$\{1.00, 0.10, 0.10, 1.00, 1.00\}$	1.0	36.0	0.79	183.6	0.0

Table 6. Different non-dominated solutions characteristics (other maps).

Map	Weights	Lowest speed	Distance	Recharge time	Consumption	Final charge rate (%)
BH	$\{0.10, \mathbf{1.00}, \mathbf{0.10}, 0.10, \mathbf{1.00}\}$	1.0	225	4.6	462.05	**100.0**
SP	$\{0.10, \mathbf{1.00}, \mathbf{0.10}, 0.10, \mathbf{1.00}\}$	1.0	59.0	1.14	120.95	**95.0**
RJ	$\{1.00, \mathbf{1.00}, \mathbf{0.10}, 1.00, \mathbf{1.00}\}$	1.0	122.0	2.49	251.05	**100.0**
BH	$\{0.10, 1.00, 0.10, 0.10, 0.10\}$	1.0	57	**0.27**	**116.85**	12.0
SP	$\{1.00, 0.10, 1.00, 1.00, 0.10\}$	1.0	55.0	**0.11**	**112.75**	0.0
RJ	$\{1.00, 0.10, 0.10, 1.00, 0.10\}$	1.0	60.0	**0.21**	**123.0**	0.0
BH	$\{\mathbf{1.00}, \mathbf{1.00}, 1.00, \mathbf{0.10}, 1.00\}$	**10.0**	**57.0**	0.4	142.5	0.0
SP	$\{\mathbf{1.00}, \mathbf{1.00}, 0.10, \mathbf{0.10}, 0.10\}$	**10.0**	**55.0**	0.35	137.5	0.0
RJ	$\{\mathbf{1.00}, \mathbf{1.00}, 0.10, \mathbf{0.10}, 0.10\}$	**10.0**	**60.0**	0.48	150.0	0.0

Notes. Bold values relate similar weights to the main objectives affected.



(A) (B)

FIGURE 11. Interactive visualization tool for checking relations between objectives – Pareto reference. (a) Lower boundary of Obj0. (b) Lower boundary of Obj1.

the concern with consumption (Green Computing) and the dynamism of this problem shows a practical approach for real applications. A MILP model was formulated taking these aspects in consideration.

A case of study was developed inspired by a real Brazilian region, using the geography of the area as baseline to the location of prohibited points. The experiments performed showed the correlation between the different objectives.

In terms of computational efficiency, metaheuristic algorithms are, in general, reasonable approaches for achieving good quality solutions in large instances. Then, future works aim to use this MILP implementation of benchmark to metaheuristic implementations as in Marques *et al.* [21].

Acknowledgements. The authors would like to thank the Brazilian agencies FAPERJ, CNPq and the Coordination for the Improvement of Higher Education Personnel – Brazil (CAPES).

REFERENCES

[1] G.J. Adabo, Long range unmanned aircraft system for power line inspection of brazilian electrical system. *J. Energy Power Eng.* **8** (2014) 394–398.

[2] N. Agatz, P. Bouman and M. Schmidt, Optimization approaches for the traveling salesman problem with drone. *Transp. Sci.* **52** (2018) 965–981.

[3] V.N. Coelho and R.H. Koochaksaraei, Non-dominated solutions for time series learning and forecasting. *Optim. Lett.* **16** (2022) 395–408.

[4] V.N. Coelho, I.M. Coelho, B.N. Coelho, M.W. Cohen, A.J.R. Reis, S.M. Silva, M.J.F. Souza, P.J. Fleming and F.G. Guimaraes, Multi-objective energy storage power dispatching using plug-in vehicles in a smart-microgrid. *Renew. Energy* **89** (2016) 730–742.

[5] V.N. Coelho, A. Grasas, H. Ramalhinho, I.M. Coelho, M.J.F. Souza and R.C. Cruz, An ils-based algorithm to solve a large-scale real heterogeneous fleet VRP with multi-trips and docking constraints. *Eur. J. Oper. Res.* **250** (2016) 367–376.

[6] B.N. Coelho, V.N. Coelho, I.M. Coelho, L.S. Ochi, D. Zuidema, M.S.F. Lima and A.R. da Costa, A multi-objective green UAV routing problem. *Comput. Oper. Res.* **88** (2017) 306–315.

[7] V.N. Coelho, I.M. Coelho, B.N. Coelho, G.C. de Oliveira, A.C. Barbosa, L. Pereira, A. de Freitas, H.G. Santos, L.S. Ochi and F.G. Guimarães, A communitarian microgrid storage planning system inside the scope of a smart city. *Appl. Energy* **201** (2017) 371–381.

[8] C. Deng, S. Wang, Z. Huang, Z. Tan and J. Liu, Unmanned aerial vehicles for power line inspection: a cooperative way in platforms and communications. *J. Commun* **9** (2014) 687–692.

[9] S. Erdoğan and E. Miller-Hooks, A green vehicle routing problem. *Transp. Res. Part E: Logistics Transp. Rev.* **48** (2012) 100–114.

[10] D. Floreano and R.J. Wood, Science, technology and the future of small autonomous drones. *Nature* **521** (2015) 460.

[11] G. Gutin and A.P. Punnen, The Traveling Salesman Problem and its Variations. Vol. 12. Springer Science & Business Media (2006).

[12] N. Haala, M. Cramer, F. Weimer and M. Trittler, Performance test on UAV-based photogrammetric data collection. *Proc. Int. Arch. Photogrammetry Remote Sensing Spatial Inf. Sci.* **38** (2011) 7–12.

[13] R. Haghnazar Koochaksaraei, F. Gadelha Guimarães, B. Hamidzadeh and S. Hashemkhani Zolfani, Visualization method for decision-making: a case study in bibliometric analysis. *Mathematics* **9** (2021) 940.

[14] A. Harris, J.J. Sluss, H.H. Refai and P.G. LoPresti, Alignment and tracking of a free-space optical communications link to a UAV. In: The 24th Digital Avionics Systems Conference, 2005. DASC 2005. Vol. 1. IEEE (2005) 1–C.

[15] I. Hong, M. Kuby and A.T. Murray, A range-restricted recharging station coverage model for drone delivery service planning. *Transp. Res. Part C: Emerg. Technol.* **90** (2018) 198–212.

[16] J. Irizarry, M. Gheisari and B.N Walker, Usability assessment of drone technology as safety inspection tools. *J. Inf. Technol. Const. (ITcon)* **17** (2012) 194–212.

[17] H.Y. Jeong, B.D. Song and S. Lee, Truck-drone hybrid delivery routing: payload-energy dependency and no-fly zones. *Int. J. Prod. Econ.* **214** (2019) 220–233.

[18] R.H. Koochaksaraei, R. Enayatifar and F.G. Guimarães, A new visualization tool in many-objective optimization problems. In: International Conference on Hybrid Artificial Intelligence Systems. Springer (2016).

[19] R.H. Koochaksaraei, I.R. Meneghini, V.N. Coelho and F.G. Guimaraes, A new visualization method in many-objective optimization with chord diagram and angular mapping. *Knowl.-Based Syst.* **138** (2017) 134–154.

[20] G. Laporte, The vehicle routing problem: an overview of exact and approximate algorithms. *Eur. J. Oper. Res.* **59** (1992) 345–358.

[21] E.L. Marques, V.N. Coelho, I.M. Coelho, B.N. Coelho and L.S. Ochi, A multi-objective metaheuristic for a green UAV grid routing problem. In: International Conference on Variable Neighborhood Search. Springer (2019) 152–166.

[22] K. Máthé and L. Buşoniu, Vision and control for UAVS: a survey of general methods and of inexpensive platforms for infrastructure inspection. *Sensors* **15** (2015) 14887–14916.

[23] I.R. Meneghini, R.H. Koochaksaraei, F.G. Guimarães and A. Gaspar-Cunha, Information to the eye of the beholder: data visualization for many-objective optimization. In: 2018 IEEE Congress on Evolutionary Computation (CEC). IEEE (2018) 1–8.

[24] N. Metni and T. Hamel, A UAV for bridge inspection: visual servoing control law with orientation limits. *Autom. Const.* **17** (2007) 3–10.

[25] N. Nigam and I. Kroo, Persistent surveillance using multiple unmanned air vehicles. In: 2008 IEEE Aerospace Conference. IEEE (2008) 1–14.

[26] S. Poikonen and J.F. Campbell, Future directions in drone routing research. *Networks* **77** (2021) 116–126.

[27] S. Poikonen and B. Golden, Multi-visit drone routing problem. *Comput. Oper. Res.* **113** (2020) 104802.

[28] H.G. Santos and T. Toffolo, Python-mip online documentation. https://python-mip.readthedocs.io (2020).

[29] B. Shneiderman, C. Plaisant, M.S. Cohen, S. Jacobs, N. Elmqvist and N. Diakopoulos, Designing the User Interface: Strategies for Effective Human-Computer Interaction. Pearson (2016).

[30] P. Vansteenwegen, W. Souffriau and K. Sörensen, The travelling salesperson problem with hotel selection. *J. Oper. Res. Soc.* **63** (2012) 207–217.

[31] X. Wang, S. Poikonen and B. Golden, The vehicle routing problem with drones: several worst-case results. *Optim. Lett.* **11** (2017) 679–697.