



HAL
open science

Using UAV-Based Systems to Monitor Air Pollution in Areas with Poor Accessibility

Óscar Alvear, Nicola Roberto Zema, Enrico Natalizio, Carlos T. Calafate

► **To cite this version:**

Óscar Alvear, Nicola Roberto Zema, Enrico Natalizio, Carlos T. Calafate. Using UAV-Based Systems to Monitor Air Pollution in Areas with Poor Accessibility. *Journal of Advanced Transportation*, 2017, 2017, pp.8204353. 10.1155/2017/8204353 . hal-01672154

HAL Id: hal-01672154

<https://hal.science/hal-01672154>

Submitted on 23 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Using UAV-based systems to monitor air pollution in areas with poor accessibility

Óscar Alvear A.*^{1,3}, Nicola Roberto Zema², Enrico Natalizio², and
Carlos T. Calafate¹

¹Department of Computer Engineering, Universitat Politècnica de
València, Camino de Vera, S/N, 46022 Valencia, Spain.

²Sorbonne Universités, Université de Technologie de Compiègne,
CNRS, Laboratoire Heudiasyc, 57 Avenue de Landshut, CS 60319,
60203 Compiègne Cedex, France

³Department of Electrical Engineering, Electronics and
Telecommunications, Universidad de Cuenca, Av. 12 de Abril, S/N,
Cuenca, Ecuador.

Abstract

Air pollution monitoring has recently become an issue of utmost importance in our society. Despite crowdsensing approaches could be an adequate solution for urban areas, they cannot be implemented in rural environments. Instead, deploying a fleet of UAVs could be considered an acceptable alternative. Embracing this approach, this paper proposes the use of UAVs equipped with off-the-shelf sensors to perform air pollution monitoring tasks. These UAVs are guided by our proposed Pollution-driven UAV Control (PdUC) algorithm, which is based on a chemotaxis metaheuristic and a local particle swarm optimization strategy. Together, they allow to automatically perform the monitoring of a specified area using UAVs. Experimental results show that, when using PdUC, an implicit priority guides the construction of pollution maps by focusing on areas where the pollutants' concentration is higher. This way, accurate maps can be constructed in a faster manner when compared to other strategies. The PdUC scheme is compared against various standard mobility models through simulation, showing that it achieves better performance. In particular, it is able to find the most polluted areas with more accuracy, and provides a higher coverage within the time bounds defined by the UAV flight time.

Keywords: Mobile sensing; pollution monitoring; spatial interpolation.

*oscar.alvear@alttics.com

1 Introduction

Industrial growth has brought unforeseen technological advances to our societies. Unfortunately, the price to pay for these advances has been an increase of air pollution worldwide, affecting both our health [1] and our lifestyle.

Air quality monitoring is relevant not only for the people living in urban areas, but also because it directly affects crops and different animals/insects in rural environments [2]. Thus, different solutions for measuring air quality should be sought for such environments.

For above mentioned reasons, environmental organizations and governmental institutions are beginning to consider the monitoring of environmental pollutants as a primary goal [3, 4].

The majority of methods used insofar to keep track of air pollution in major cities rely on fixed monitoring stations [5]. However, the use of such dedicated architectures and hardware for pollution monitoring is outmatched, in theory, by the use of *crowdsensing* [6] in areas with a high population density. Also, new ground-vehicle-based mobile sensors, which would theoretically be able to cover the same areas as the fixed solutions while employing a reduced number of agents, are emerging as a viable alternative (e.g. [7, 8]).

With respect to the widespread use of small pollution monitoring sensors embedded in mobile vehicles, the possible scenarios can be divided into two main classes:

- **Urban environments**, where it is possible to embed the sensors on a wide set of vehicles like bicycles [9, 10] or cars [8];
- **Rural and industrial areas**, where vehicular traffic is scarce and limited to the main transportation arteries.

In the latter case, crowdsensing often fails to provide enough data to obtain realistic measurements having the required granularity.

Regarding crowdsensing approaches, projects like [11, 12, 13] relied on crowdsensing solutions to monitor pollution in urban areas. However, in rural and industrial zones, available options are quite more limited. In the particular case of large rural or industrial areas, a fleet of mobile vehicles could be efficiently used to cover the vast distances associated with them. Furthermore, the use of autonomous sensor carriers is even more encouraged in this case due to the following considerations:

- The relative absence of civilian population to be taken care of during robotic operations
- Stable and regulated positioning of obstacles
- Fewer constraints concerning UAV flight laws

- Safety and security concerns, as some areas could be dangerous to access for human operators.

Since, in these environments, ground access is usually hindered and full of obstacles, the most feasible way to implement a fleet of mobile pollution-monitoring robots is via Unmanned Aerial Vehicles (UAVs) [14].

Taking the aforementioned issues into consideration, in this paper we propose the use of UAVs equipped with commercial and off-the-shelf (COTS) devices and sensors to implement a service of air pollution monitoring that leverages the use of bio-inspired approaches as its main control strategy. These choices allow covering a specific area automatically and enable discovering the pollution distribution of a large area by prioritizing the most polluted zones inside it.

We show that, using our chemotaxis-based approach for UAV path control, it is possible to achieve faster and more accurate estimations about the location of the most polluted areas with respect to classical area-search approaches. Our analysis also takes into account uncertainty-based considerations in the sensor sampling operations.

This paper is organized as follows: in section 2 we refer to some related works addressing UAV-based sensing, UAV mobility models, and UAV control protocols. Sections 3 and 4 present an overview of the UAV Configuration and the UAV Control System, respectively. In section 5, we compare our algorithm against the Billiard and Spiral mobility models via simulation. Section 6 discusses the open issues in air pollution monitoring using UAVs. Finally, in section 7, we present the conclusions of our work.

2 Related Works

UAV-based solutions have experienced a very substantial increase in the last decade, especially in the past five years. Back in 2004, NASA experts defined a wide set of civil applications for UAVs [15], highlighting their potential in the near future in areas such as commercial, Earth Sciences, national security, and land management. This preliminary report was ratified years later by authors such as Hugenholtz et al. [16], who explained how the use of UAVs could revolutionize research methods in the fields of Earth Sciences and remote sensing. In [17], authors display the results of a detailed study on different UAVs aspects, showing their applicability in Agriculture and Forestry, Disaster Monitoring, Localization and Rescue, Surveillance, Environmental Monitoring, Vegetation Monitoring, Photogrammetry, and so on.

If we focus specifically on research using quadrotor multi-copters, authors like Gupte et al. [18] and Colomina y Molina [19] consider that, given their high maneuverability, compactness and ease of use, different applications for these devices are being found in areas including civil engineering, search and

rescue, emergency response, national security, military surveillance, border patrol and surveillance, as well as in other areas such as Earth Sciences, where they can be used to study climate change, glacier dynamics, volcanic activity, or for atmospheric sampling, among others.

In our case, we are more interested in atmospheric sampling to measure air pollution levels. In this research area, Anderson and Gaston [20] highlight the applicability of UAVs in the field of ecology, emphasizing that the spatial and temporal resolutions of the data obtained by traditional methods often fail to adapt well to the requirements of local ecology-oriented research. Furthermore, the use of UAVs, when flying at low altitudes and speeds, offers new opportunities in terms of ecological phenomena measurements, enabling the delivery of data with a finer spatial resolution. In fact, Zhang and Kovacs [21] explain how the images taken by small UAVs are becoming an alternative to high-resolution satellite images, which are much more expensive, to study the variations in crop and soil conditions. Specifically, the use of UAVs is considered a good alternative given its low cost of operation in environmental monitoring, its high spatial and temporal resolution, and its high flexibility in the scheduling of image acquisitions. A good example of this use can be found in the work of Bellvert et al. [22], which shows how, by using a multicopter equipped with a thermal camera, it was possible to obtain a very precise map of water levels in a vineyard, thereby achieving significant advances in the field of precision agriculture.

Focusing on our topic, despite the presence of several works related to air pollution monitoring using Unmanned Aerial Systems (UAS), the majority of these involve, mainly, swarm creation or communication interaction between them. An example of such work is [23], where authors propose a mobility model for a group of nodes following "Virtual Tracks" (highways, valley, etc.) operating in a predefined "Switch Station" mode, through which nodes can split or merge with another group of nodes.

Different works have been done related to mounting sensors in Unmanned Aerial Vehicles. In this regard, [24] et al. use an UAV equipped with a sensor to create a Wireless Sensor Network, thereby enabling each UAV to act as a sink or as a node, but it does not try to optimize the monitoring process. [25] et al. propose a fixed-wind aircraft carrying a sensor node that acts as a mobile gateway, showing the communication between the UAV and different static base stations which monitor the pollution. In this case, the UAV only recovers the data collected by the stations. [26] et al. propose the design of a lightweight laser-based sensor for measuring trace gas species using UAVs, mainly analyzing how the optical sensor captures the air pollution samples. In [27], authors use a large-sized aircraft equipped with ozone sensors to cover a wide area in an automated manner, showing how the UAV improves the sampling granularity.

If we analyze works related to mobility models for UAS mobility control that could be used for air pollution monitoring tasks, we can observe that

basically no work focuses on the coverage improvement for a certain area.

For instance, in [28], authors propose a mobility model based on the Enhanced Gauss-Markov model to eliminate or limit the sudden stops and sharp turns that the random waypoint mobility model typically creates. Also, in [29], authors present a semi-random circular movement (SRCM) based model. They analyze the coverage and network connectivity by comparing results against a random waypoint mobility model.

The authors of [30] compare their models against random waypoint-based, Markov-based, and Brownian-motion-based algorithms to cover a specific area, analyzing the influence of the use of collision avoidance systems in the time to achieve a full area coverage. The work in [31] compares the results of using the "Random Mobility Model" and the "Distributed Pheromone Repel Mobility Model" as direction decision engines (next waypoint) in UAV environments. The authors of [32] propose an algorithm to cover a specific area; it selects a point in space along with the line perpendicular to its heading direction and then drives the UAV based on geometric considerations.

There are works focusing on using UAVs for specific tasks involving autonomous movements. An example is [33], where authors present a mobility model for the self-deployment of an Aerial Ad Hoc Network in a disaster scenario in order to create a flying and flexible communications infrastructure that victims can use. The mobility model proposed is based mainly on the Jaccard dissimilarity metric to control the deployment of the Unmanned Aerial Vehicles composing the network. A similar work is presented in [34], where instead an in-network density analysis is used to select the physical areas that need to be visited by a flying robot.

Focusing solely on existing proposals addressing mobility models, we can find works such as [35] where authors propose the Paparazzi Mobility Model (PPRZM) by defining five types of movements - Stay-On, Way-Point, Eight, Scan and Oval - following a defined state machine with different probabilities to change between states. There are even studies following animal-based navigation patterns. An example of such work is [36], where authors investigate the UAV placement and navigation strategies with the end goal of improving network connectivity, using local flocking rules that aerial living beings like birds and insects typically follow.

The use of UAVs for air pollution monitoring in a specific area using multi-rotor drones is, however, still not present in scientific literature, and this work can be seen as one of the first approaches in this direction. Our contribution can be divided into two parts: (i) the design of a low-cost and open-source UAV equipped with off-the-shelf sensors for monitoring tasks, and (ii) the deployment of a protocol called PdUC (Pollution-driven UAV Control) to automatically track a target area by focusing on the most polluted regions.

3 Overview of the proposed solution

To implement a solution for air pollution monitoring using UAVs we have to consider, like in any cyber-physical system, two main aspects: (i) the hardware configuration, and (ii) the control process for controlling the system behavior.

By following these guides, our proposal can also be split into two parts: (i) the physical configuration of the UAV and the environmental sensors, and (ii) the algorithm to control the UAV for automatically monitoring a specific area called Pollution-driven UAV Control - PdUC.

Even though we are not proposing, in this paper, an implementation using real UAVs, we nevertheless present the specification of the devised cyber-physical system.

3.1 UAV Configuration

We have designed a scheme to dynamically drive the UAV by connecting the UAV control module to a Raspberry Pi [37], and connecting the latter to the set of pollution sensors via an analog converter. The scheme is shown in figure 1.

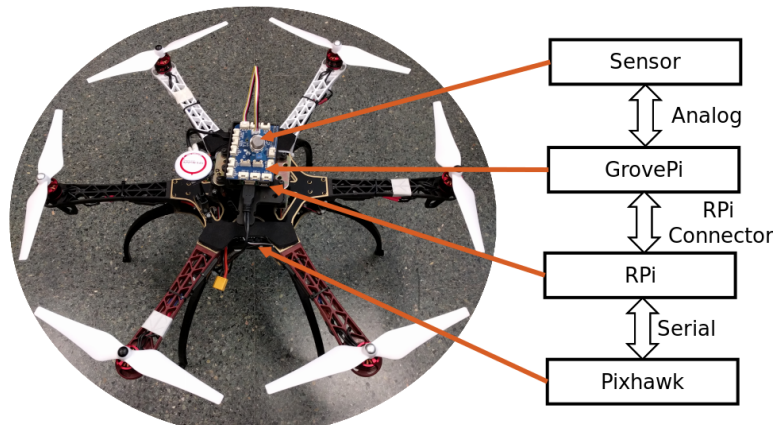


Figure 1: Proposed UAV with air pollution sensors.

The UAV is driven using a Pixhawk Autopilot [38, 39], which controls its physical functioning. The Raspberry Pi is mounted over the UAV chassis and connected to the Pixhawk through a serial port. The sensors are connected to the Raspberry Pi using a Grove Raspberry Hat (GrovePi) [40], which allows connecting different kinds of COTS sensors easily. Specifically, we are using:

- **Pixhawk Autopilot:** a high-performance flight control module suitable for several types of autonomous vehicles including multi-rotors,

helicopters, cars, boats, and fixed-wing aircrafts. It is developed under the independent, open hardware Pixhawk project, and it has two main components: (i) an Autopilot hardware provides an industry standard autopilot module designed as a 168 MHz Cortex M4F CPU with 3D ACC/Gyro/MAG/Baro sensors, microSD slot, 5 UARTs, CAN, I2C, SPI, ADC, etc.; (ii) an Autopilot software that includes a real-time Operating System (RTOS) with a POXIS-style environment to control the drone.

- **Raspberry Pi:** one of the most popular single-board-computers (SBC) worldwide. It is a low-cost and small-sized piece of hardware that allows exploring computing, and that supports different Operating Systems. The most popular of them is Raspbian, which is based on Debian, although Ubuntu Mate or Windows 10 IoT Core can also be installed, thereby allowing to use several programming languages. Besides, all Raspberry Pi versions benefit from several input/output ports operating at 5V, thus being ideal for all sorts of IoT projects.
- **GrovePi:** extension board that allows connecting several analog/digital *grove* ports to a Raspberry Pi in an easy way. It has several Grove ports: seven digital ports, three analog ports, three I2C ports, one serial port to the GrovePi, and a serial connector to the Raspberry Pi.
- **Grove Sensors:** sensors which use a grove-standardized connector, providing an easy connection to different boards like GrovePi. There are several COTS environmental sensors such as CO₂, CO or Alcohol, among others. Specifically, we mostly focus on Ozone sensors (MQ131).

Figure 2 shows the closed-loop control scheme of our proposal. The Pixhawk autopilot is responsible for the physical control system of the UAV (Lower level), while the Raspberry Pi is in charge of the Guidance system (Higher Level) determining the way forward.

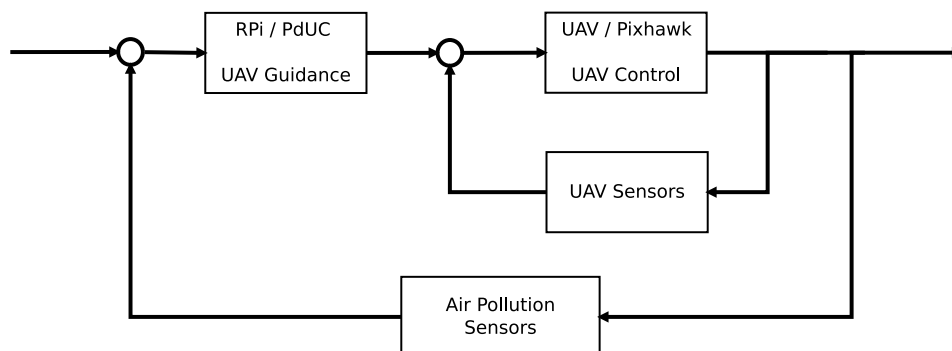


Figure 2: UAV control loop.

3.2 Autonomous Driving

To deploy an algorithm for automatically monitoring a specific area we have analyzed, first of all, different existing possibilities that could be useful to our goals.

So, to elaborate the proposed PdUC solution, we have used specific techniques such as the metaheuristics and optimization algorithms described below.

3.2.1 Chemotaxis meta-heuristic

The use of rotary-wing UAVs, equipped with chemical sensors and tasked to survey large areas, could follow chemotactic [41] mobility patterns, since their flight behavior could easily implement the following two-phase algorithm: first, read a pollution concentration while hovering; next, follow a chemotactic step.

Chemotaxis meta-heuristics are based on bacteria movement. In this model, the microorganisms react to a chemical stimulus by moving towards areas with a higher concentration of some components (e.g. food) or moving away from others (e.g. poison). In our system, we have considered the following adaptation of the chemotaxis. Let us consider an agent i moving on a Euclidean plane, located at position \vec{P}_j^i from an absolute reference axis, and moving along time in sequential steps j . For every chemotactic step, a new position \vec{P}_j^i is calculated based on the previous one, defined by x_{j-1}^i and y_{j-1}^i , plus a step size d^i applying a random direction θ_j^i , as specified in equation 1.

$$\vec{P}_j^i = \begin{pmatrix} x_{j-1}^i \\ y_{j-1}^i \end{pmatrix} + \begin{pmatrix} d^i \times \cos(\theta_j^i) \\ d^i \times \sin(\theta_j^i) \end{pmatrix} \quad (1)$$

$$\theta_j^i = \begin{cases} \theta_{j-1}^i + \alpha_j^i & p_j^i \geq p_{j-1}^i \\ -\theta_{j-1}^i + \beta_j^i & p_j^i < p_{j-1}^i \end{cases} \quad (2)$$

The direction θ_j^i , as shown in equation 2, is calculated on the basis of the concentration value of a certain chemical component, sampled by an agent i at step j : p_j^i . With respect to the previously sampled value p_{j-1}^i , the following two types of movements are contemplated: *Run* and *Tumble*. In the former, Run, when the component concentration is increased with respect to the previous sample, the movement continues to follow the same direction as before (θ_{j-1}^i) plus a random angle α_j^i . Regarding the latter, Tumble, when the concentration is decreasing, the movement takes a turn in the opposite direction $-\theta_{j-1}^i$, plus a random angle β_j^i . Notice that both α_j^i and β_j^i are used to introduce variability and to maximize the gradient, allowing to reach the most polluted areas faster.

3.2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a technique introduced in [42] where a solution to a problem is represented as a particle p^i moving in a D-dimensional space at a time t ; each particle p^i maintains its position p_t^i , and its best performance position p_b^i . To determine the next position p_{t+1}^i , PSO calculates the stochastic adjustment in the direction of the previous local best position of i 's p_b^i element, along with the general best position of any element p_b^g , as shown in equation 3:

$$p_{t+1}^i = \alpha \cdot p_t^i + U(0, \beta) \cdot (p_b^i - p_t^i) + U(0, \beta) \cdot (p_b^g - p_t^i) \quad (3)$$

where α and β are a constants to calibrate the algorithm, and $U(0, \beta)$ is a random number between $[0, \beta]$.

4 Proposed Autonomic Solution

To consistently drive the UAVs, so as to achieve the desired area coverage goals, we have devised the following algorithm, which incorporates a chemotactic approach.

4.1 PdUC Algorithm

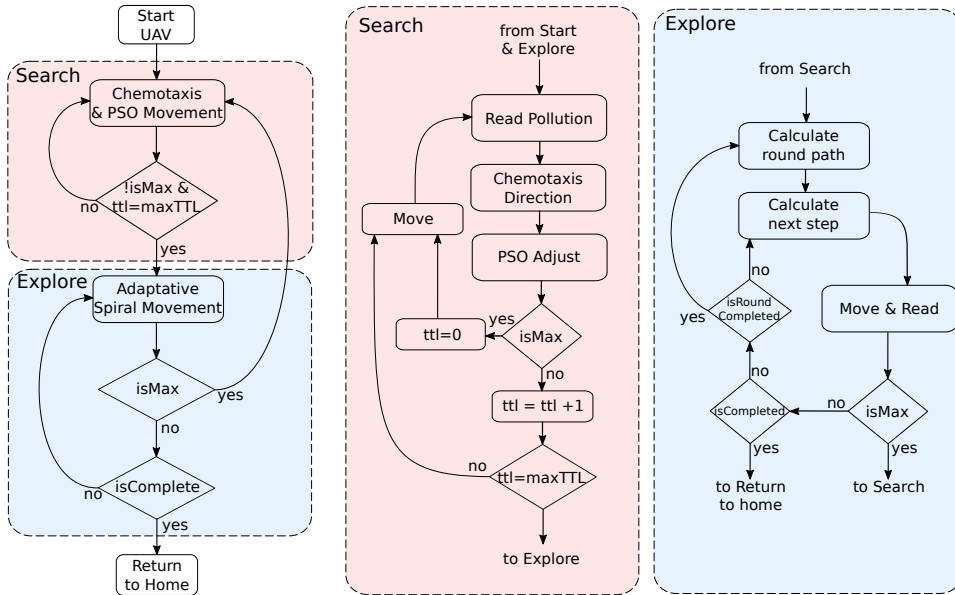


Figure 3: Overview of different states associated to the PdUC algorithm.

In this context, we have developed an algorithm called Pollution-driven UAV Control (PdUC), based on the chemotaxis metaheuristic concept, to

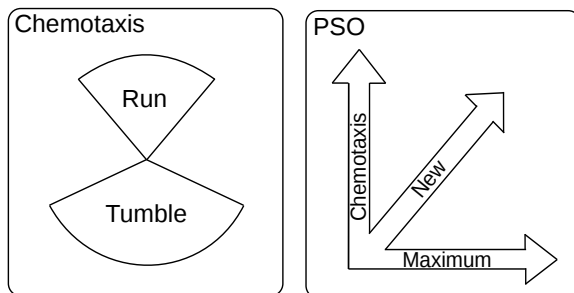


Figure 4: PdUC algorithm: calculation of a new direction.

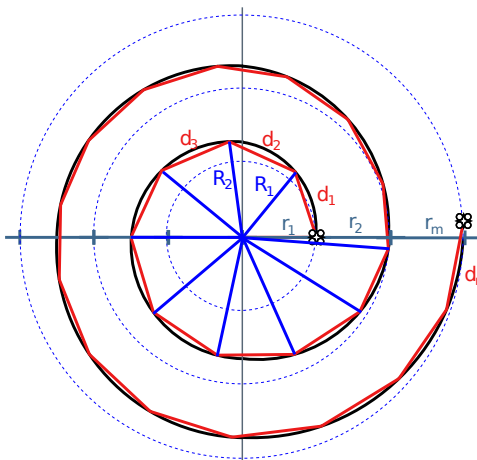


Figure 5: PdUC algorithm: exploration phase.

search an area for the highest pollution concentration levels. Once this pollution hotspot is found, the flying drone covers the whole area by following a spiral movement, starting from the most polluted location.

Our algorithm is composed of two phases: (i) A search phase, in which the UAV searches for a globally maximum pollution value, and (ii) An exploration phase, where the UAV explores the surrounding area, following a spiral movement, until one of the following conditions occurs: it covers the whole area, the allowed flight time ends, or it finds another maximum value, in which case it returns to the search phase, as shown Figure 3.

The exploration phase is based mainly on two previously described techniques: a chemotaxis metaheuristic and a local Particle Swarm Optimization algorithm. As detailed in Algorithm 1, initially, before the UAV starts its first movement, it samples the pollution value and puts it in a buffer. For each chemotactic step, it starts to hover, collects another sample, and compares it with the previous one. If the sampling variation is positive (increasing), the UAV follows a "Run" chemotaxis direction, with a random α_j^i of $[-30, 30]$ degrees. Otherwise, if the sampling variation is decreasing, the

UAV calculates the "Tumble" chemotaxis direction in the reverse orientation with a random β_j^i of $[-150, 150]$ degrees, although modified by the actual maximum value reached (md^i), as shown in figure 4. Equation 4 denotes the formula to calculate the new direction, and γ specifies the weight of the md^i , which must be between 0 and 1.

Algorithm 1 PdUC Search Phase.

```

1: while isSearching do
2:    $p_2^{pollution} \leftarrow CurrentPollution(p_2)$ 
3:    $\nabla poll \leftarrow p_2^{pollution} - p_1^{pollution}$ 
4:    $p_1 \leftarrow p_2$ 
5:   if  $\nabla poll > 0$  then
6:      $tll \leftarrow 0$ 
7:      $p_2 \leftarrow Run(p_1)$ 
8:      $p_{max} \leftarrow p_2$ 
9:   else
10:     $p_2 \leftarrow Tumble(p_1)$ 
11:     $tll \leftarrow tll + 1$ 
12:     $p_2 \leftarrow AdjustPSO(p_2, p_{max})$ 
13:    if  $isInsideArea(p_2)$  then
14:       $MoveTo(p_2)$ 
15:    else
16:       $p_2 \leftarrow Tumble(p_1)$ 
17:    if  $tll > tll_{max}$  then
18:       $isSearching \leftarrow false$ 
19:       $isExploring \leftarrow true$ 
20: end

```

To determine when PdUC has found a maximum local value, we use a TTL (Time-to-live) counter. When PdUC finds a maximum value, the TTL is reset and increasing until a new maximum pollution value is found, or until the maximum TTL value is reached. In this case, PdUC reverts to the exploration phase since it considers that a local maximum value has been found.

$$\theta_j^i = \begin{cases} \theta_{j-1}^i + \alpha_j^i & \text{Run} \\ (1 - \gamma)(-\theta_{j-1}^i + \beta_j^i) + \gamma md^i & \text{Tumble} \end{cases} \quad (4)$$

Once a maximum value is reached, the next phase is to explore the surrounding area. As shown in algorithm 2, this is achieved by following an Archimedean spiral similar to the one depicted in Figure 5. Starting from the maximum value, it covers the surrounding area by applying a basic step size d_j^i , and changing it depending on the detected pollution variations, a procedure that is similar to the *finding* phase. If the variation is less than

a preset value c^i , the step size increases until reaching $3 \times d_j^i$; otherwise, it decreases until d_j^i is reached. If a maximum pollution value is found, PdUC automatically returns to the exploration phase. Finally, once the whole area is covered, the UAV changes to a *return-to-base* (RTB) mode to finish the exploration.

Algorithm 2 PdUC Exploration Phase.

```

1: while isExploring do
2:    $round \leftarrow round + 1$ 
3:    $round_{size} \leftarrow 2\pi \cdot \frac{round + round_{next}}{2}$ 
4:    $round_{direction} \leftarrow -previous_{direction}$ 
5:    $angle_{count} \leftarrow \frac{2\pi}{\frac{round_{size}}{d}}$ 
6:    $step \leftarrow 0$ 
7:    $angle \leftarrow 0$ 
8:   while  $step < round_{size}$  and isExploring do
9:     if isInsideArea( $p_2$ ) and isNotMonitored( $p_2$ ) then
10:       $p_2^{pollution} \leftarrow CurrentPollution(p_2)$ 
11:      if  $p_2 > p_{max}$  then
12:         $store(p_{max})$ 
13:         $store(round)$ 
14:         $isExploring = false$ 
15:         $isSearching = true$ 
16:         $p_{max} = p_2$ 
17:      else
18:         $\nabla poll \leftarrow p_2^{pollution} - p_1^{pollution}$ 
19:         $MoveTo(p_2)$ 
20:       $step \leftarrow step + d$ 
21:       $angle \leftarrow angle + angle_{count} \times round_{direction}$ 
22:       $p_1 \leftarrow p_2$ 
23:       $p_2 \leftarrow NextPoint(p_1, angle, step)$ 
24:       $previous_{direction} \leftarrow round_{direction}$ 
25: end

```

4.2 Algorithm optimization

Next, analyzing the overall behavior, we have introduced some modifications to optimize the performance of the proposed PdUC algorithm.

Spiralling with alternating directions

As shown in Figure 6, to avoid large steps in the exploration phase when the spiral center is next to a border, the direction of the spiral will alternate for each round to allow minimizing the length of some of the steps. To this

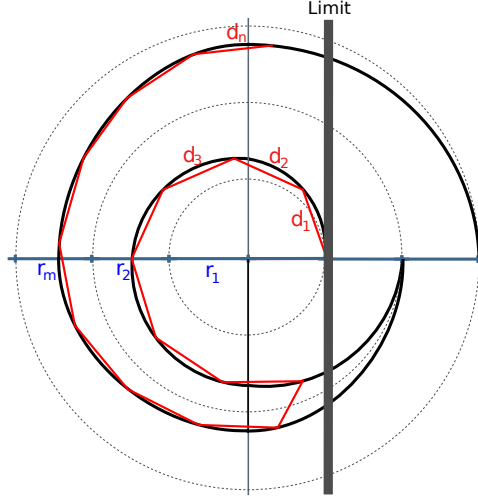


Figure 6: PdUC algorithm: alternating spiral direction.

purpose, for each spiral round, we calculate the direction adopted as being the opposite direction with reference to the previously used one. The system can get the general size of the area to search, as well as its borders, before starting the mission. This procedure takes place in line 4 of Algorithm 2. In detail, it follows equations 5 and 6:

$$\theta_{s,r} = \begin{cases} \alpha + \beta_{s,c} & \text{if } r \text{ is even} \\ \alpha - \beta_{s,c} & \text{if } r \text{ is odd} \end{cases} \quad (5)$$

$$p_{s,r} = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} x_c + R_s \times \cos(\theta_{s,r}) \\ y_c + R_s \times \sin(\theta_{s,r}) \end{pmatrix} \quad (6)$$

Where $\theta_{s,r}$ defines the angle in round r and step s , α is the initial angle, and β_s is the angle in step s . Using it, angle $\theta_{s,r}$ and the next point p_s are calculated using as a reference the coordinates for the spiral center (x_c and y_c) and radius R_s .

Skipping Previously Monitored Areas

As shown in Figure 7, to avoid monitoring the same area multiple times, all samples, which were taken within the area monitored during the exploration phase, are internally stored. For this purpose, PdUC maintains a list containing the location of the central position of all spirals with their respective radius to determine the monitored areas (as a circumference determined by a center and a radius). Next, in the exploration phase, all points inside these circles are omitted for the sake of celerity, as shown in line 9 of Algorithm 2.

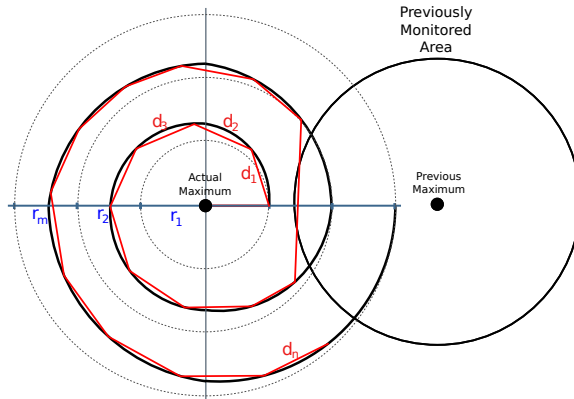


Figure 7: PdUC algorithm: skipping monitored areas

5 Validation & Simulation

To validate our protocol, we have run several simulations with different configurations implemented in the OMNeT++ simulation tool, as shown in figure 8.

To prepare a suitable data environment, we have created various pollution distribution maps representing ozone levels to be used as inputs for testing. These pollution maps were generated using the *R Graph* tool [43], and following a Kriging-based interpolation [44]. In particular, a Gaussian distribution is used to adjust the parameters coming from random data sources of ozone concentration. The actual values range between 40 and 180 ppb, thereby providing a realistic ozone distribution.

Figure 9 shows some samples of the created maps, which have the highest pollution concentration (areas in red) located at completely different positions due to the stochastic scenario generation procedure adopted.

Using the previously created data as input, we have run several simulations using OMNeT++, comparing our protocol against both the Billiard and Spiral mobility patterns. In the simulator, we have created a mobility model implementation of PdUC. In addition, to simulate the sampling process, we have configured OMNeT++ to periodically perform measurements taken from the pollution distribution map defined for the test.

Figure 10 shows an example of the path followed by an UAV using the PdUC algorithm as a guidance system. As expected, the UAV starts a search process throughout the scenario until it locates a position with the highest degree of pollution (local maximum). Afterward, it follows a spiral pattern to gain awareness of the surrounding gradients. If, while following the spiral-shaped scan path, it finds a higher pollution value, the algorithm again switches to the search phase. Finally, when the entire target area has been sampled, the algorithm finishes.

To compare the three options under study, we recreate, using the R

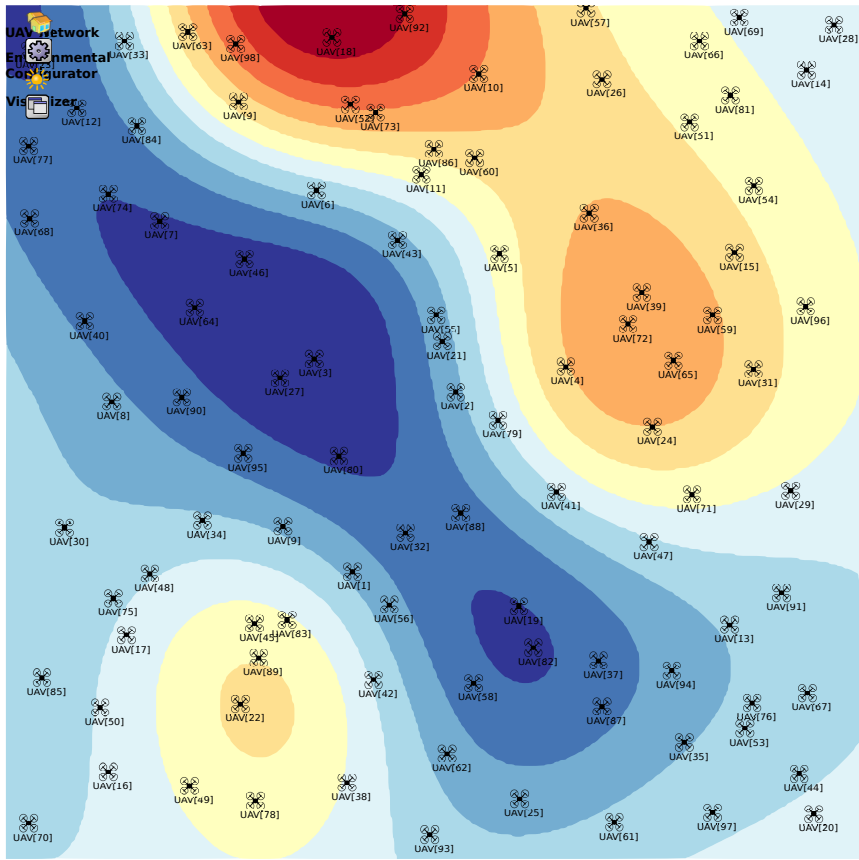


Figure 8: Example simulation scenario showing possible initial UAV positions over a randomly generated pollution map.

Graph tool, the pollution distribution maps using the simulation output as the input for the Kriging-based interpolation. In this way, we obtain new pollution maps for comparison against the ones used as reference.

Table 1 summarizes the parameters used in the simulations.

Table 1: Simulation parameters.

Parameter	Value
Area	4x4 Km
Pollution range	[40 - 180] ppb
Sampling error	10 ppb
Max. speed	20 m/s
Sampling time	4 seconds
Step distance	100 m
Mobility models	Billiard, Spiral and PdUC

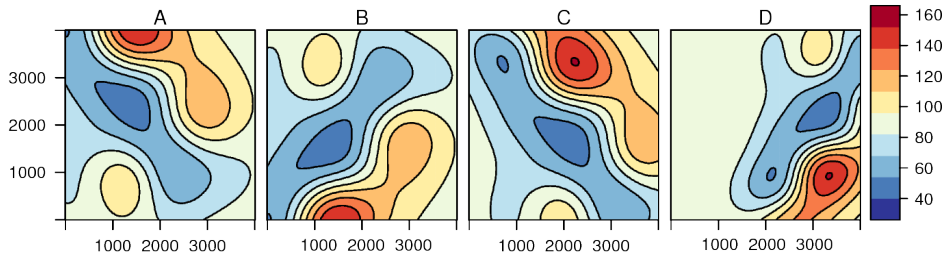


Figure 9: Pollution distribution examples used for validation.

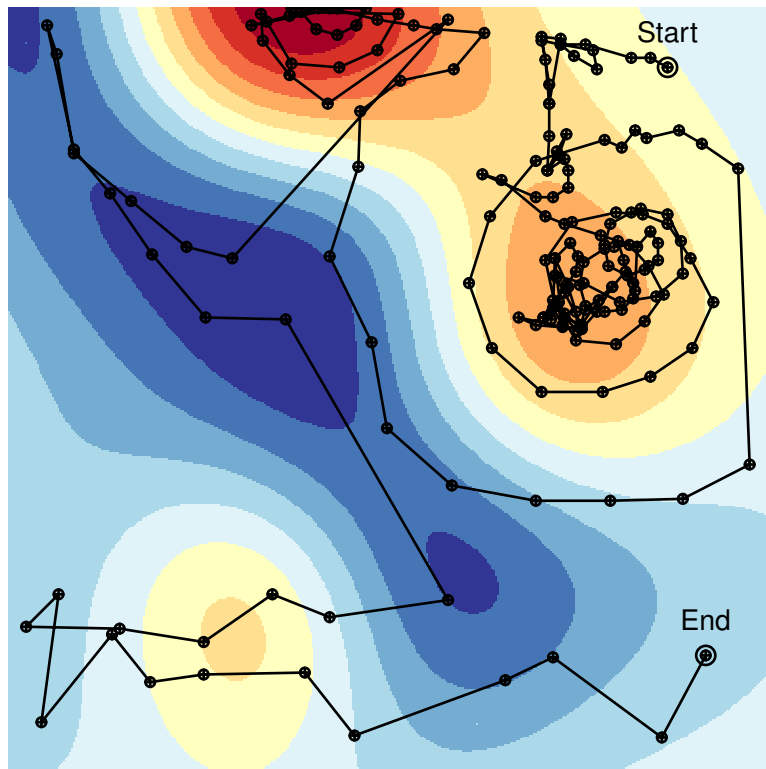


Figure 10: Example of an UAV path when adopting the PdUC mobility model.

Since we are proposing the PdUC algorithm for rural environments, the simulation area defined is a 4×4 Km area. As indicated above, the pollution distribution relies on synthetic maps that are generated by combining a random Kriging interpolation following a Gaussian model with values between 40 and 180 units based on the Air Quality Index (AQI) [45]. Since samples are taken using off-the-shelf sensors, which are not precise, we introduce a random sampling error of ± 10 ppb based on real tests using the MQ131 (Ozone) sensor. In our simulation, we set the maximum UAV speed

to 20 m/s, a value achievable by many commercial UAVs. The step distance defined between consecutive samples is 100 meters. Once a new sampling location is reached, the monitoring time per sample is defined to be 4 seconds.

The mobility models used are Billiard, Spiral, and PdUC. These models have different assumptions regarding the initial UAV position. In the *Billiard* model, the UAV starts in a corner of the target area, and then covers the whole area by "bouncing" when reaching the borders. The *Spiral* model starts at the center of the area to cover, and then gradually moves to the periphery of the scenario following a spiral pattern. Finally, PdUC is set to start at a random position within the target area.

We now proceed by analyzing the time required to cover the entire area using each of the approaches being tested. For this purpose, we defined 100 simulations for each model (Billiard, Spiral, and PdUC), and determined the required time to cover the whole area, estimating the pollution map afterward.

For each run, the starting position of the UAV is randomly set on the map, as shown in Figure 8.

Figure 11 shows the Cumulative Distribution Function relative to the time required to cover the whole area for the three mobility models. It can be seen that the Billiard and Spiral models do not depend on the start position, spending a nearly constant time (5600 and 2600 seconds, respectively) for each configuration defined. In the case of the PdUC mobility model, since it reacts to air pollution, the time required to cover the complete area varies between 1800 and 4300 seconds, depending on the start position.

Due to battery restrictions, it is interesting to analyze how fast each mobility model discovers the most polluted areas, and how accurately does it recreate the pollution distribution. For this purpose, we analyze the relative error for the three mobility models at different time instants (600, 1200, 1800, 2400, 3000 and 6000 seconds); this error is defined by equation 7:

$$e_t = \frac{\sum_{i=1}^m \sum_{j=1}^n \left| \frac{s_{x,y,t} - b_{x,y}}{\Delta b} \right|}{m \cdot n} \quad (7)$$

where, e_t is the relative error at time t ; $s_{x,y,t}$ is the recreated pollution value at position (x, y) using the samples taken during simulation until time t , $b_{x,y}$ is the reference pollution value at position (x, y) , and n and m are the dimensions of the target area, respectively.

Figure 12 shows the temporal evolution of the relative error between the three mobility models (Billiard, Spiral, and PdUC) and the original one. We can observe that all mobility models have roughly the same behavior: they start with a high relative error, which is foreseeable since we are using Kriging interpolation to recreate the pollution distribution, and it tends to the mean value when the number of samples is not enough. Then, as more samples become available, the spatial interpolation process quickly becomes

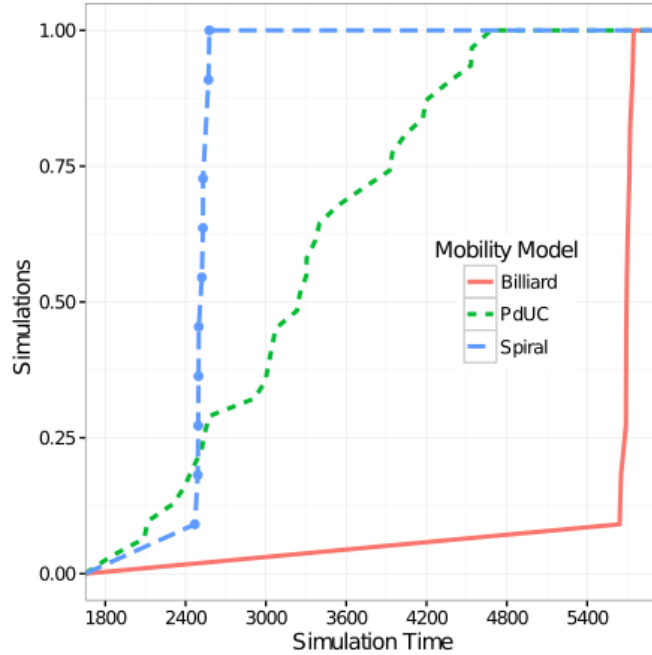


Figure 11: Cumulative Distribution Function of the time spent at covering the complete area for the Billiard, Spiral and PdUC mobility models.

more precise.

Although the three mobility models are similar, the spiral approach achieves a better performance in terms of relative error reduction. However, if we analyze only the most polluted regions, that is, regions characterized by values higher than a certain threshold (120 and 150 ppm in our case, based on AQI [45]), we find that PdUC clearly provides better results.

Figures 13 and 14 show the comparison between the Billiard, Spiral, and PdUC mobility models at different times when only focusing on air pollution values higher than 120 and 150 ppb, respectively. These results show that PdUC clearly provides better results than the Billiard and Spiral movement patterns, outperforming their accuracy from nearly the beginning of the experiment (1200 seconds), and reaching the lowest relative error values in just 3600 seconds, with these two other mobility approaches more than doubling the error values for a same time. In particular, the Billiard mobility pattern requires about 6000 seconds to achieve a similar degree of accuracy (120 ppb case), while the Spiral approach is not able to achieve values as low as PdUC in any of the cases. This occurs because PdUC focuses on the highest values in the chemotaxis-based phase. PdUC always prioritizes the most polluted areas in detriment of less polluted ones, thus allowing to obtain, at least, details about the region with the highest pollution values.

To complete our study, figure 15 presents an example of the evolution

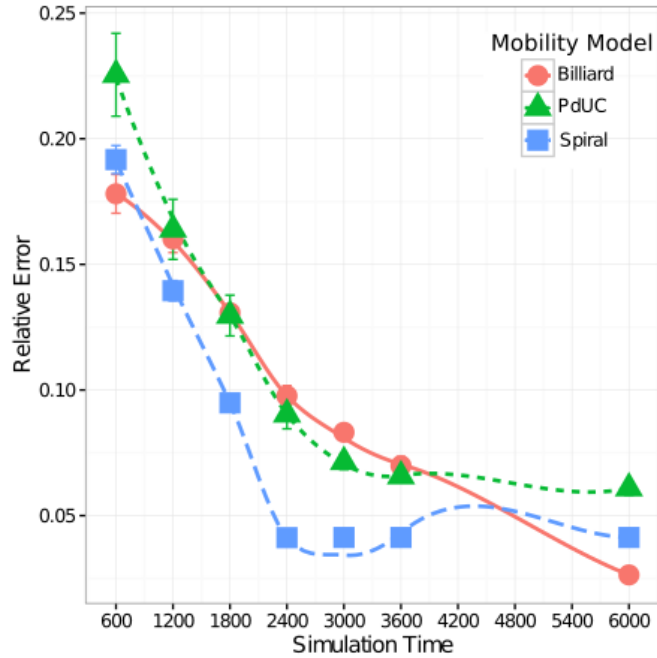


Figure 12: Relative error comparison between the PdUC, Billiard, and Spiral mobility models at different times, when analyzing all the values.

of predicted pollution values for the whole target area, and for the three algorithms under analysis (Spiral, Billiard, and PdUC), at different times (1200s, 2400s, 3600s and 6000s). We can observe that PdUC is able to quickly find the most polluted areas, while the effectiveness of other approaches highly depends on the actual location of pollution hotspots in order to detect them at an early stage.

6 Open Issues

Unmanned Aerial Systems (UAS) have been quickly adopted in different application areas due to their flexibility and relatively low cost. Focusing in the environmental monitoring area, in a previous work we introduced the idea of using UAVs for air pollution monitoring [46] by equipping them with off-the-shelf sensors. Instead, in the current paper, we introduce an algorithm called PdUC to guide a single UAV in the task of monitoring a specific area. However, there are still several open issues related to this topic.

Until now, we have only considered operations limited to a single UAV. The next step in our research is to introduce multiple-UAVs and the associated cooperation schemes. The following aspects need to be addressed to follow this research line:

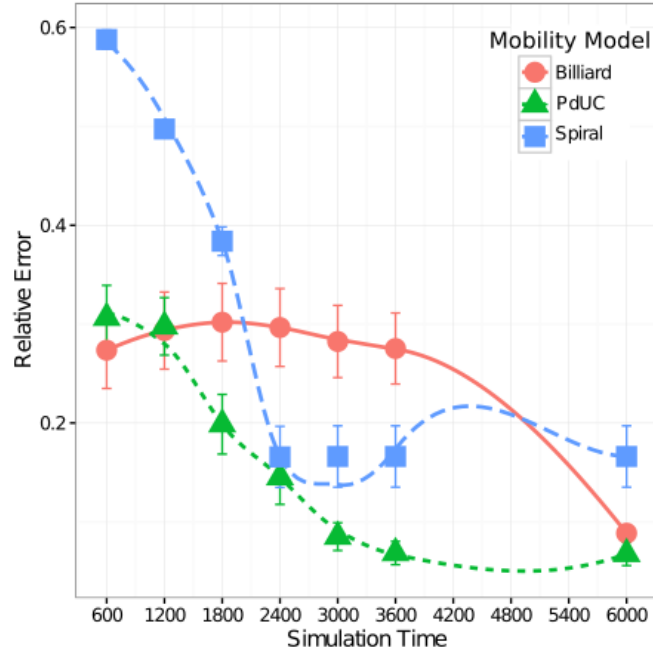


Figure 13: Relative error comparison between PdUC, Billiard, and Spiral mobility models at different times when only considering values higher than 120 ppb.

- **Cooperation:** to maximize the effectiveness and reduce mapping times it is advisable to have several UAVs that cooperate with each other to achieve a same task, thereby accelerating the whole process, and avoiding battery exhaustion before completing the monitoring process.
- **Collision Avoidance:** since the different UAVs are expected to have some degree of autonomy regarding their mobility pattern, a correct coordination between nearby UAVs is required to avoid collisions when flying at a close range.
- **Communications:** to achieve the aforementioned goals of cooperation and collision avoidance, communications between UAVs, and between UAVs and a central management unit, are required.

On the other hand, using mobile sensors installed on UAVs introduces new issues to the sensing process that should also be addressed:

- **Altitude:** despite currently most pollution studies are made at a ground level, the use of UAVs allows determining the concentration of pollutants at different heights, allowing to determine if there are layers of pollutants that can cause health problems in rugged mountainsides.

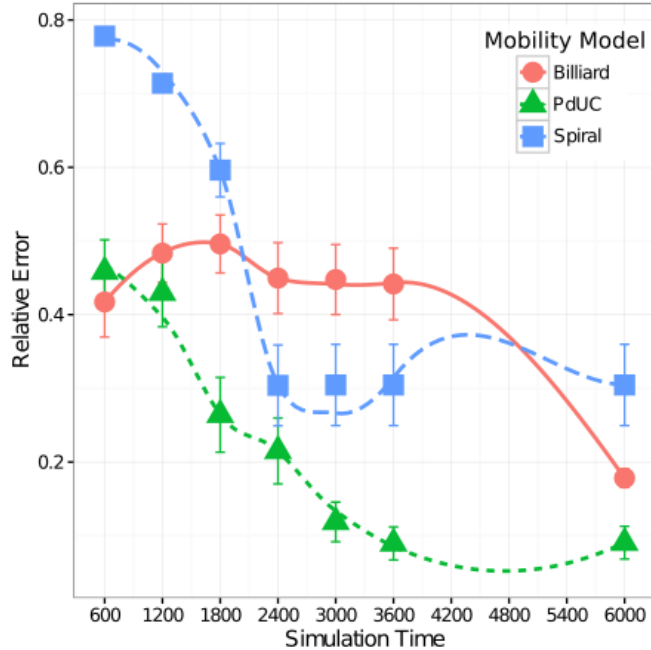


Figure 14: Relative error comparison between PdUC, Billiard, and Spiral mobility models at different times when only considering values higher than 150 ppb.

- **Influence of the wind:** the sampling procedure includes sensors that are sensitive to the wind conditions. In addition, wind causes the overall pollution map to be more dynamic. In this context, both issues deserve more scrutiny.

7 Conclusions

Despite we have several options to monitor air pollution in urban scenarios, being crowdsensing an emerging approach arousing great interest, finding an adequate approach for industrial or rural areas remains a pending task.

Recently, Unmanned Aerial Systems have experienced unprecedented growth, offering a platform for the fast development of solutions due to their flexibility and relatively low cost; in fact, they can be a good option to solve the previous requirements, allowing to monitor remote areas that are difficult to access.

In this paper, we propose a solution where we equip an UAV with off-the-shelf sensors for monitoring tasks, using a Pixhawk Autopilot for UAV control, and a Raspberry Pi for sensing and storing environmental pollution data.

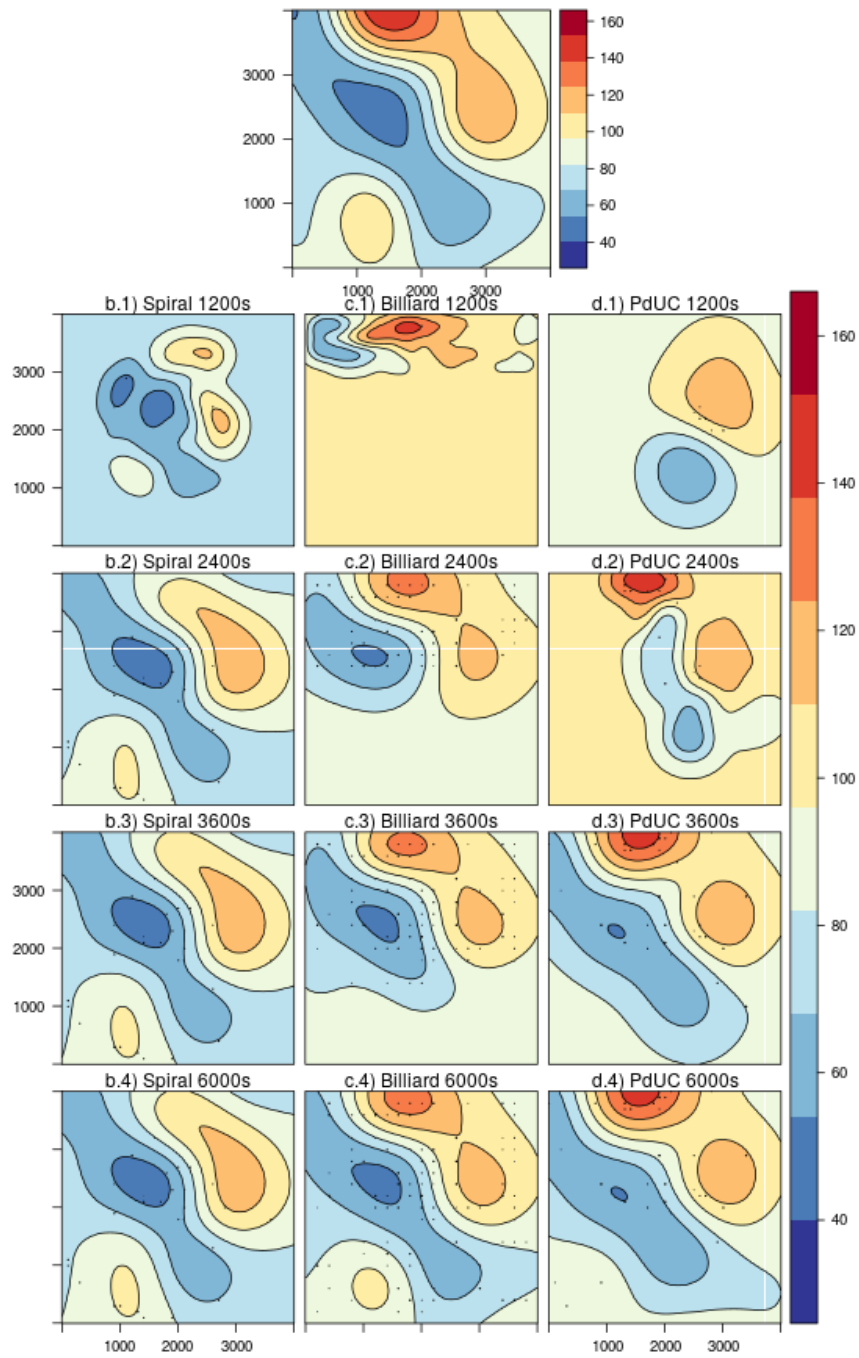


Figure 15: Visual representation of the estimation output for the PdUC, Billiard, and Spiral mobility models at different times.

To automatically analyze pollution values within a target area, we also propose an adaptive algorithm for autonomous navigation called Pollution-based UAV Control system (PdUC). This algorithm allows an UAV to autonomously monitor a specific area by prioritizing the most polluted zones. In particular, PdUC combines different concepts including a Chemotaxis metaheuristic, a local Particle Swarm Optimization (PSO), and an Adaptive Spiralling technique, to create an algorithm able to quickly search for hotspots having high pollution values, and to cover the surrounding area as well, thereby obtaining a complete and detailed pollution map of the target region.

To validate our proposal, we compared the proposed PdUC solution against the Billiard and Spiral mobility models through simulations implemented in OMNeT++. Simulation experiments show that PdUC offers significantly better performance at reducing prediction errors, especially regarding the accuracy achieved for the high-values range.

Acknowledgment

This work has been partially carried out in the framework of the DIVINA Challenge Team, which is funded under the Labex MS2T program. Labex MS2T is supported by the French Government, through the program "Investments for the future" managed by the National Agency for Research (Reference: ANR-11-IDEX-0004-02). This work was also supported by the "Programa Estatal de Investigación, Desarrollo e Innovación Orientada a Retos de la Sociedad, Proyecto I+D+I TEC2014-52690-R", the "Programa de becas SENESCYT de la República del Ecuador", and the Research Direction of University of Cuenca. The received funding do not lead any conflict of interests regarding the publication of the manuscript.

References

- [1] A. Seaton, D. Godden, W. MacNee, K. Donaldson, Particulate air pollution and acute health effects, *The lancet* 345 (8943) (1995) 176–178.
- [2] Q. McFrederick, J. Kathilankal, J. Fuentes, Air pollution modifies floral scent trails, *Atmospheric Environment* 42 (10) (2008) 2336 – 2348.
- [3] D. Mage, G. Ozolins, P. Peterson, A. Webster, R. Orthofer, V. Vandeweerd, M. Gwynne, Urban air pollution in megacities of the world, *Atmospheric Environment* 30 (5) (1996) 681–686.
- [4] H. Mayer, Air pollution in cities, *Atmospheric environment* 33 (24) (1999) 4029–4037.

- [5] P. S. Kanaroglou, M. Jerrett, J. Morrison, B. Beckerman, M. A. Arain, N. L. Gilbert, J. R. Brook, Establishing an air pollution monitoring network for intra-urban population exposure assessment: A location-allocation approach, *Atmospheric Environment* 39 (13) (2005) 2399–2409.
- [6] O. Alvear, W. Zamora, C. Calafate, J.-C. Cano, P. Manzoni, An architecture offering mobile pollution sensing with high spatial resolution, *Journal of Sensors* 2016.
- [7] A. Adam-poupart, A. Brand, M. Fournier, M. Jerrett, A. Smargiassi, Spatiotemporal modeling of ozone levels in Quebec (Canada): a comparison of kriging, land-use regression (LUR), and combined Bayesian Maximum Entropy and LUR approaches, *Environmental Health Perspectives* 970 (January 2013) (2014) 1–19. doi:10.1289/ehp.1306566.
- [8] M. Pujadas, J. Plaza, J. Teres, B. Artalejo, M. Millan, Passive remote sensing of nitrogen dioxide as a tool for tracking air pollution in urban areas: the madrid urban plume, a case of study, *Atmospheric Environment* 34 (19) (2000) 3041–3056.
- [9] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, A. T. Campbell, Bikenet: A mobile sensing system for cyclist experience mapping, *ACM Transactions on Sensor Networks (TOSN)* 6 (1) (2009) 6.
- [10] M. André, The artemis european driving cycles for measuring car pollutant emissions, *Science of the total Environment* 334 (2004) 73–84.
- [11] M. Brković, V. Sretović, Urban sensing—smart solutions for monitoring environmental quality: Case studies from serbia, in: 48th ISOCARP International Congress: Fast Forward: Planning in a (hyper) dynamic urban context, Perm, Russia, 2012.
- [12] S. C. Hu, Y. C. Wang, C. Y. Huang, Y. C. Tseng, Measuring air quality in city areas by vehicular wireless sensor networks, *Journal of Systems and Software* 84 (11) (2011) 2005–2012. doi:10.1016/j.jss.2011.06.043.
- [13] Y. Cheng, X. Li, Z. Li, S. Jiang, Y. Li, J. Jia, X. Jiang, AirCloud: a cloud-based air-quality monitoring system for everyone, in: Proceedings of the 12th ACM Conference on Embedded Network Sensor Systems - SenSys '14, ACM, 2014, pp. 251–265. doi:10.1145/2668332.2668346. URL <http://dl.acm.org/citation.cfm?doid=2668332.2668346>
- [14] M. Dunbabin, L. Marques, Robots for environmental monitoring: Significant advancements and applications, *IEEE Robotics & Automation Magazine* 19 (1) (2012) 24–39.

- [15] T. H. Cox, C. J. Nagy, M. A. Skoog, I. A. Somers, R. Warner, Civil uav capability assessment.
- [16] C. H. Hugenholtz, B. J. Moorman, K. Riddell, K. Whitehead, Small unmanned aircraft systems for remote sensing and earth science research, *Eos, Transactions American Geophysical Union* 93 (25) (2012) 236–236.
- [17] G. Pajares, Overview and current status of remote sensing applications based on unmanned aerial vehicles (uavs), *Photogrammetric Engineering & Remote Sensing* 81 (4) (2015) 281–329.
- [18] S. Gupte, P. Mohandas, J. Conrad, A survey of quadrotor Unmanned Aerial Vehicles, in: *Southeastcon, 2012 Proceedings of IEEE, 2012*, pp. 1–6.
- [19] I. Colomina, P. Molina, Unmanned aerial systems for photogrammetry and remote sensing: A review, *ISPRS Journal of Photogrammetry and Remote Sensing* 92 (2014) 79–97.
- [20] K. Anderson, K. Gaston, Lightweight unmanned aerial vehicles will revolutionize spatial ecology, *Frontiers in Ecology and the Environment* 11 (3) (2013) 138 – 146.
- [21] C. Zhang, J. Kovacs, The application of small unmanned aerial systems for precision agriculture: a review, *Precision Agriculture* 13 (6) (2012) 693–712.
- [22] J. Bellvert, P. Zarco-Tejada, J. Girona, E. Fereres, Mapping crop water stress index in a 'pinot-noir' vineyard: comparing ground measurements with thermal remote sensing imagery from an unmanned aerial vehicle, *Precision Agriculture* 15 (4) (2014) 361–376.
- [23] B. Zhou, K. Xu, M. Gerla, Group and swarm mobility models for ad hoc network scenarios using virtual tracks, in: *Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE, Vol. 1, IEEE, 2004*, pp. 289–294.
- [24] A. T. Erman, L. van Hoesel, P. Havinga, J. Wu, Enabling mobility in heterogeneous wireless sensor networks cooperating with uavs for mission-critical management, *IEEE Wireless Communications* 15 (6) (2008) 38–46.
- [25] S. K. Teh, L. Mejias, P. Corke, W. Hu, Experiments in integrating autonomous uninhabited aerial vehicles(uavs) and wireless sensor networks, in: *2008 Australasian Conference on Robotics and Automation (ACRA 08), The Australian Robotics and Automation Association Inc., Canberra, 2008*.
URL <https://eprints.qut.edu.au/15536/>

- [26] A. Khan, D. Schaefer, L. Tao, D. J. Miller, K. Sun, M. A. Zondlo, W. A. Harrison, B. Roscoe, D. J. Lary, Low power greenhouse gas sensors for unmanned aerial vehicles, *Remote Sensing* 4 (5) (2012) 1355–1368.
- [27] S. Illingworth, G. Allen, C. Percival, P. Hollingsworth, M. Gallagher, H. Ricketts, H. Hayes, P. Áadosz, D. Crawley, G. Roberts, Measurement of boundary layer ozone concentrations on-board a Skywalker unmanned aerial vehicle, *Atmospheric Science Letters* 15 (4) (2014) 252–258.
- [28] J.-D. M. M. Biomo, T. Kunz, M. St-Hilaire, An enhanced gauss-markov mobility model for simulations of unmanned aerial ad hoc networks, in: *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP, IEEE, 2014*, pp. 1–8.
- [29] W. Wang, X. Guan, B. Wang, Y. Wang, A novel mobility model based on semi-random circular movement in mobile ad hoc networks, *Information Sciences* 180 (3) (2010) 399–413.
- [30] D. Orfanus, E. P. de Freitas, Comparison of uav-based reconnaissance systems performance using realistic mobility models, in: *2014 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), IEEE, 2014*, pp. 248–253.
- [31] E. Kuiper, S. Nadjm-Tehrani, Mobility models for uav group reconnaissance applications, in: *2006 International Conference on Wireless and Mobile Communications (ICWMC'06), IEEE, 2006*, pp. 33–33.
- [32] Y. Wan, K. Namuduri, Y. Zhou, S. Fu, A smooth-turn mobility model for airborne networks, *IEEE Transactions on Vehicular Technology* 62 (7) (2013) 3359–3370.
- [33] S. Toral, D. Reina, F. Barrero, et al., A self organising aerial ad hoc network mobility model for disaster scenarios, in: *Developments of E-Systems Engineering (DeSE), 2015 International Conference on, IEEE, 2015*, pp. 35–40.
- [34] O. Briante, V. Loscri, P. Pace, G. Ruggeri, N. R. Zema, COMVIVOR: An Evolutionary Communication Framework Based on Survivors' Devices Reuse, *Wireless Personal Communications* 85 (4) (2015) 2021–2040. doi:10.1007/s11277-015-2888-y.
URL <http://dx.doi.org/10.1007/s11277-015-2888-y>
- [35] O. Bouachir, A. Abrassart, F. Garcia, N. Larrieu, A mobility model for uav ad hoc network, in: *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on, IEEE, 2014*, pp. 383–388.

- [36] P. Basu, J. Redi, V. Shurbanov, Coordinated flocking of uavs for improved connectivity of mobile ground nodes, in: Military Communications Conference, 2004. MILCOM 2004. 2004 IEEE, Vol. 3, IEEE, 2004, pp. 1628–1634.
- [37] R. P. Foundation, Raspberry pi, <https://www.raspberrypi.org/>, Accessed: October 10, 2015.
- [38] L. Meier, P. Tanskanen, F. Fraundorfer, M. Pollefeys, Pixhawk: A system for autonomous flight using onboard computer vision, in: Robotics and automation (ICRA), 2011 IEEE international conference on, IEEE, 2011, pp. 2992–2997.
- [39] L. Meier, P. Tanskanen, L. Heng, G. H. Lee, F. Fraundorfer, M. Pollefeys, Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision, *Autonomous Robots* 33 (1) (2012) 21–39. doi:10.1007/s10514-012-9281-4.
URL <http://dx.doi.org/10.1007/s10514-012-9281-4>
- [40] D. Industries, Grovepi, <http://www.dexterindustries.com/grovepi/>, Accessed: October 10, 2015.
- [41] I. Boussaid, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, *Information Sciences* 237 (2013) 82–117.
- [42] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, IEEE, 1995, pp. 39–43.
- [43] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2016).
URL <http://www.R-project.org/>
- [44] M. L. Stein, *Statistical Interpolation of Spatial Data: Some Theory for Kriging*, Springer, New York, 1999.
- [45] U. S. E. P. Agency, Air Quality Index Available: <http://cfpub.epa.gov/airnow/index.cfm?action=aqibasics.aqi> (2015).
- [46] O. Alvear, C. T. Calafate, E. Hernández, J.-C. Cano, P. Manzoni, Mobile pollution data sensing using uavs, in: *Proceedings of the 13th International Conference on Advances in Mobile Computing and Multimedia*, ACM, 2015, pp. 393–397.