IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 June 2022, accepted 29 June 2022, date of publication 4 July 2022, date of current version 8 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3188204

RESEARCH ARTICLE

Threat Management Methodology for Unmanned Aerial Systems Operating in the U-Space

CARLOS CAPITÁN, JESÚS CAPITÁN, ÁNGEL R. CASTAÑO, AND ANÍBAL OLLERO, (Fellow, IEEE)
GRVC Robotics Laboratory, University of Seville, 41092 Seville, Spain

Corresponding author: Carlos Capitán (ccapitan@us.es)

This work was supported by the European Union's Horizon 2020 Research and Innovation Program through the Galileo-EGNOS as an Asset for UTM Safety and Security (GAUSS) Project under Grant 776293.

ABSTRACT This paper presents a threat management methodology for *Unmanned Aircraft Systems* (UAS) operating in the civil airspace. The work is framed within an *Unmanned Traffic Management* (UTM) system based on the U-space initiative. We propose a new method that focuses on providing the required automated decision-making during real-time threat management and conflict resolution, which is one of the main gaps in the current U-space ecosystem. Our method is capable of handling all commonplace UTM threats, as well as selecting optimal mitigation actions, trading off efficiency and safety. Our implementation is open-source and fully integrated in a UTM software architecture, implementing U-space services related to emergency management and tactical deconfliction. We demonstrate our methodology through a set of realistic use cases with actual UAS operating in civil airspace. For that, we performed field experiments in an aerodrome with segregated airspace, and we showcased that the methodology is capable of autonomously managing heterogeneous threats in real time.

INDEX TERMS UTM, U-space, UAS, threat management.

I. INTRODUCTION

Although they have been extensively used for military purposes for a long time, *Unmanned Aircraft Systems* (UAS), or drones, are becoming lately quite popular for commercial and civil applications. Indeed, some sources [1] predict a total market value of 10 billion euros per year by 2035, with up to 400,000 drones providing services in the airspace by 2050. Among this wide variety of civil drone applications, we can find examples like last-mile delivery [2], surveillance [3], infrastructure inspection [4], traffic monitoring [5], media production [6], or health emergency situations [7].

As a consequence, interested parties have started to regulate and restructure the *Very Low Level* (VLL) civil airspace (below 120 *m*) to meet UAS requirements and integrate them [8]. In fact, UAS integration into civil airspace may be one of the most revolutionary events since the appearance of *Air Traffic Management* (ATM). Traditionally, ATM has been handled by means of voice communication through a centralized *Air Traffic Control* entity. However, the rise of UAS operations at a large scale makes it necessary to redesign the paradigm for airspace management, in order to achieve a better scalability and distribute responsibilities among different stakeholders and actors. There are already relevant initiatives for the integration of UAS into the VLL civil airspace. On the one hand, the *National Aeronautics and Space Administration* (NASA) created the concept for *UAS Traffic Management* (UTM) [9], to enable safe, large-scale operations with UAS in low-altitude airspace [10]. On the other hand, Europe has recently extended this UTM concept by proposing the *U-space* ecosystem [11].

In this context, the *European Union* (EU) has recently published a new regulatory framework for UAS [12], [13]. This regulation establishes three risk-based categories for UAS operations: (i) an *Open* category, which only includes low-risk operations with no involvement of the *National Aviation Authority* (NAA); (ii) an *Specific* category that requires the submission of a risk assessment to the NAA for operation

The associate editor coordinating the review of this manuscript and approving it for publication was Xujie Li.

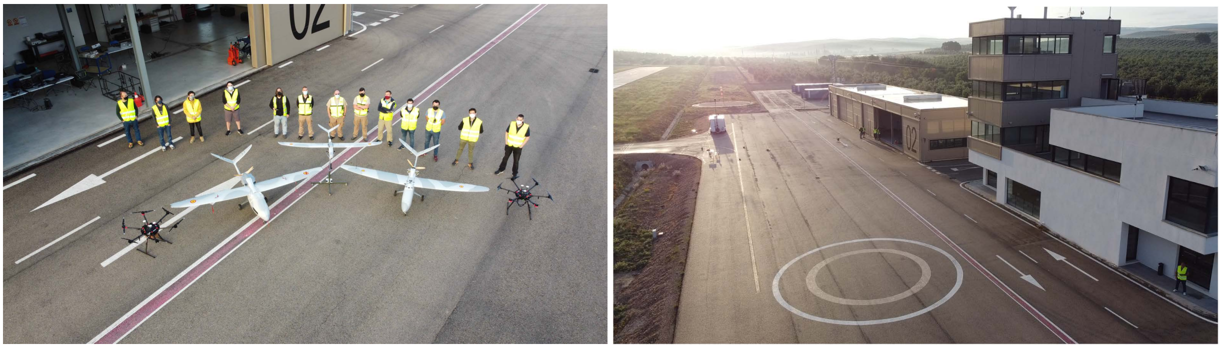


FIGURE 1. Left, the team of fixed- and rotary-wing UAS used to demonstrate the approach for threat management. Right, the ATLAS test facilities (Spain) where the experiments were carried out.

approval [14]; and (iii) a *Certified* category, which is for operations with risks at the level of classic manned aviation. Moreover, three additional EU regulations [15]–[17], which will be applicable from 2023, establish the framework for safe drone and manned aircraft operations in the U-space. These regulations introduce new services for drone operators, allowing them to carry out more complex and longer-distance operations, particularly in *Beyond-Line-Of-Sight* (BLOS) and congested traffic conditions. Besides, the *Joint Authorities for Rulemaking on Unmanned Systems* (JARUS),[1] made up of various NAAs and airspace stakeholders, is trying to converge to a common procedure for risk identification and assessment in UAS operations. In this sense, JARUS has published a procedure called *Specific Operational Risk Assessment* (SORA) [18], which can be used to evaluate potential risks and propose mitigation actions for UAS operations within the Specific category.

The work in this article has been developed within the context of the European project GAUSS,[2] whose main objective was leveraging the high-performance positioning functionalities provided by the Galileo ecosystem for U-space operations. In these U-space operations, while UAS are flying, unexpected events or threats might occur, leading to dangerous situations. Therefore, as a result of the GAUSS project, we proposed a generic software architecture [19] for autonomous threat management in U-space, providing the required U-space services for tactical deconfliction and real-time decision-making. In [19], we described the overall architecture with the different U-space services involved, as well as preliminary simulated experiments for validation. In this article, we extend our previous work by describing the specifics of a novel methodology for autonomous threat management in multi-UAS operations, fully integrated within the aforementioned U-space software architecture. Furthermore, we implement and demonstrate our methodology on actual fixed- and rotary-wing UAS (see Figure 1), through an experimental campaign in the ATLAS Test Centre.[3]

The main contributions of this work are as follows. First, we introduce the main concepts of the U-space initiative linked with threat management (Section II-A), and we review other relevant works about emergency management and multi-UAS conflict resolution (Section II-B). Second, we propose a new methodology for threat management in multi-UAS operations in the U-space (Section III). After providing a general overview (Section III-A), we identify a generic set of threats that can occur during UAS operations in the U-space (Section III-B). Then, our methodology is based on proposing a set of mitigation actions that are evaluated in terms of cost and risk level, in order to take optimal decisions in an autonomous fashion (Section III-C). Our approach is flexible enough to accommodate additional types of threats or mitigation actions in the future, and it has been implemented as open-source software for the community. Third, we demonstrate our methodology for threat management in real flight tests, integrated within a U-space framework (Section IV). We defined a set of use cases for multi-UAS operations involving the different types of threats identified by GAUSS (Section IV-A); and we validated our methodology through a series of field trials in the ATLAS Test Centre (Sections IV-B and IV-C). Finally, we draw the main conclusions of this work and point at future lines for further development (Section V).

II. BACKGROUND

In this section, we introduce the concepts of the U-space initiative, mainly those associated with threat management, including a review of related work on emergency management and conflict resolution methodologies.

A. U-SPACE

U-space is a collaborative effort among researchers, industry, and regulators to enable the integration of UAS operations within the VLL civil airspace, providing UAS situational awareness and digital communication with manned aviation, ATM service providers, and legal authorities. There exists a roadmap to deploy U-space in Europe, consisting of the four phases depicted in Figure 2. Each phase will propose a new set of services with increasing complexity and integration

[1] http://jarus-rpas.org

[2] https://projectgauss.eu

[3] http://atlascenter.aero/

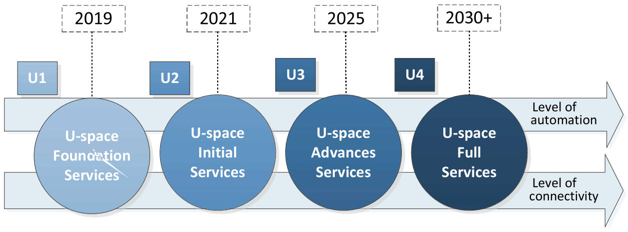


FIGURE 2. The implementation roadmap for the U-space initiative [11], consisting of 4 deployment phases.

levels between UAS and manned aircraft, as well as an upgraded version of the existing services in the previous phases.

The U-space core consists of a UTM system with software modules that provide services to the involved actors. The complete list of defined services can be consulted in [20], and their current level of development in [21]. U-space services can be classified in those that play a role in the preflight operation phase or those activated during flight:

- *Preflight* services involve the functionalities required to prepare and schedule a UAS operation. The aircraft and the operator need to register (*E-registration*), and the initial flight plan has to be handled before being accepted (*Flight planning management*). Then, the pilot may get assistance through information about predefined restricted areas (*Pre-tactical geofencing*) and the resolution of possible conflicts before flying (*Strategic deconfliction*).
- *In-flight* services involve the functionalities required to handle the operation after the UAS has taken off. This includes updates for the operator (*Tactical geofencing*) or the UAS itself (*Dynamic geofencing*) regarding geofences during the flight. Also, tracking information about the current position and predicted trajectory for each UAS (*Tracking*). This information is then used to create a situation of the airspace (*Monitoring*) and to generate warnings and contingency actions under possible threats (*Emergency management*). In order to keep a safety distance between aircraft and geofences, alternative plans could also be suggested in-flight (*Tactical deconfliction*).
- Last, there are some services that can be activated both in the preflight or in-flight phase. These services provide identification (*E-identification*), weather forecasts (*Weather Information*) or more generic information (*Drone Aeronautical Information Management*), create an interface with the ATC (*Procedural Interface with ATC* and *Collaborative interface with ATC*), or control and manage the UAS density in the airspace (*Dynamic Capacity Management*).

In this paper, we propose a methodology for autonomous decision-making in real-time threat management. This means an implementation of the Emergency Management and Tactical Deconfliction services defined in U-space. According to [21], these services have not been fully addressed by current UTM systems, existing a notorious implementation gap. In particular, these services belong to the U2 and U3 implementation phases, which are scheduled between 2021 and 2029.

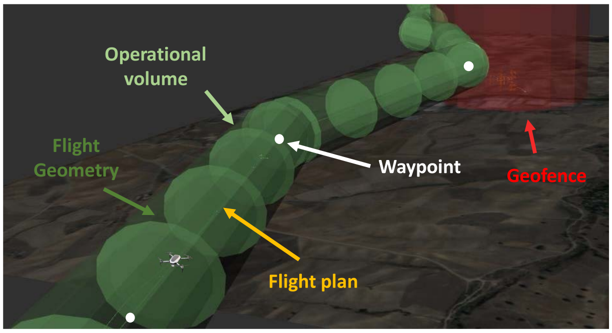


FIGURE 3. Graphical representation of the operational volume of a UAS operation. Given a flight plan (white points), the green cylindrical volume around represent its OV, including the Flight Geometry (darker green) and the Contingency Volume (lighter green). An example geofence volume is represented in red.

Another relevant concept in U-space is the definition of the *Operational Volume* (OV) of a UAS operation (See Figure 3). The OV is a 4-dimensional space that consists of a single or multi-segmented 3-dimensional volume around the flight plan, with a temporal component representing the time and duration that the volume will be occupied, as part of an operation. The OV indicates the intent of an operator to perform an operation and maintain the aircraft within the bounds of the volume at all times. After the OV for a proposed UAS operation is established, this volume stays reserved to ensure a safe operation. Given the temporal component of an OV, geographic overlapping is allowed between operations as long as they are separated in time. The OV is composed by: the *Flight Geometry*, which defines the volume of airspace where the UAS is intended to remain during its operation; and the *Contingency Volume*, which is an outer surrounding volume to account for environmental or performance uncertainties.

B. RELATED WORK FOR THREAT MANAGEMENT

Even though we focus on UAS, the threats commonly handled by manned aircraft and *Remotely Piloted Aerial Systems* (RPAS) can be taken as a good starting point to study unexpected events in the airspace and how to manage them. In [22], the characteristics of airborne conflict occurrences are detailed. Manned aviation controller guidelines to handle emergency situations are also presented in [23]. Moreover, authors in [24] describe a set of common RPAS specific emergency situations and derive corresponding contingency measures whenever feasible. They study the usability of existing procedures and standards coming from manned aviation, and then they extend some cases to unmanned aviation or RPAS (e.g., electrical failure and navigational failure). Regarding UAS, the EU-funded project CORUS[4]

[4] https://www.sesarju.eu/projects/corus

defined a new approach for the threat concept framed in the U-space. A threat is considered an unexpected event that may happen and cause harm when UAS are operating in the U-space. This project has published an exhaustive list of possible threats and events that may happen during a UAS operation in U-space [25].

In the literature, there are multiple works that address threat management in the airspace for UAS, but mainly focusing on particular types of threats. They can be split into two main categories: (i) approaches to cope with emergency events that cause malfunctioning UAS; and (ii) those dealing with conflict resolution, with other vehicles or no-fly zones. Within the first category, the authors in [26] propose some procedures to address certain events of malfunctioning UAS (e.g., a motor failure, a GPS failure, or a loss of orientation). The specific failure of loss of command and control communication link is considered in [27]. In [28], a structured approach to classify contingency sources and select contingency reactions depending on the severity is developed. Also, many works manage these malfunctioning situations by means of emergency landing operations [29]–[35]. For example, works in [29], [30] center on landing operations in the case of unpowered UAS. The detection of safe landing zones can be done using machine learning techniques [31] or vision-based approaches [33]. An overview of automated emergency landing systems can be found in [32]. In [33], it is presented a guidance, navigation, and control method for an automated emergency landing system with a fixed-wing UAS. Harmsel *et al.* [34] propose a meta-level emergency landing planner to calculate safe paths for small UAS when low-energy reserves are detected unexpectedly while flying over populated urban environments. Moreover, an emergency management architecture has also been presented for piloted or autonomous aircraft in [35]. They design and implement an adaptive flight planner that dynamically computes feasible flight plans in response to events that degrade aircraft performance.

Regarding the second category, conflict resolution can be approached in a pre-flight (strategic deconfliction) or in-flight phase (tactical deconfliction). Pre-flight solutions usually formulate the problem as multi-agent path finding. The authors in [36] propose a priority-based method and a negotiation method to solve this problem in a distributed manner, assuming the existence of multiple U-space service providers. In [37], a heavily constrained urban airspace with a high density of UAS traffic is tackled. They apply the one-way street concept plus heading/altitude rules to segment the airspace, and delay- and speed-based actions to resolve conflicts. An approach for the *Dynamic capacity management* service defined in U-space is implemented in [38]. Pre-flight UAS planning is enhanced with a dynamic reconfiguration algorithm, to balance airspace allocation by rescheduling alternative trajectory options to route away from possible congested areas.

In terms of in-flight conflict resolution, see-and-avoid [39] and velocity-obstacle methods [40] have been traditionally used for UAS collision avoidance. More recently, the use of 4D bubbles for conflict management has been applied in the U-space [41]. Moreover, a method to predict conflicts and adapt the velocity vectors to avoid them has been proposed [42]. This method has also been extended [43] to consider the operation priorities established in the U-space policy. Other works present high-level architectures for U-space. In [44], a software modular design is introduced, but not focusing on decision-making procedures. The authors in [45] do implement a U-space architecture for conflict resolution, focusing on functionalities which involve the use of vehicle-to-vehicle and vehicle-to-infrastructure technologies for communication between UAS and operators.

Most of the previous works address threat management focusing on specific types of threats, either emergency situations or inter-vehicle conflicts. The main contribution of our work is proposing a holistic methodology that considers multiple kinds of threats and mitigation actions in an integrated decision-making procedure. We selected a set of common threats and mitigation actions in UAS airspace operations, but the framework is general enough to accommodate additional ones in the future. A second contribution is that our framework is integrated within the U-space initiative. On the one hand, the considered threats, mitigation actions, and airspace constraints, are in line with those defined in the U-space. On the other hand, our implementation is based on the actual U-space services and has been tested within a software architecture replicating them [19]. Last but not least, most of the existing works, except for the one in [45], only provide results in simulation. There have also been recent field trials to test conflict resolution procedures in civil airspace [46]. However, this work reproduces predefined maneuvers instead of performing real-time decision-making. Instead, we demonstrate our methodology working on illustrative use cases with actual UAS and an actual U-space software architecture. In summary, we take ideas from previous works to define sets of relevant threats and mitigation actions in U-space operations, and then we develop a generic framework for autonomous real-time threat management. Our methodology accepts threats of multiple types and, according to a certain categorization, feasible mitigation actions are evaluated to make optimal decisions.

III. METHODOLOGY FOR THREAT MANAGEMENT

In this section, we describe our methodology for threat management in the U-space. First, we provide an overview of the problem and our solution. Then, we identify and provide a description of the threats that may occur during multi-UAS operations in the U-space. Finally, we describe our decision-making procedure to apply the corresponding mitigation actions.

A. OVERVIEW

We consider a set of UAS operating in a common airspace, where each operation has a predefined *priority* established

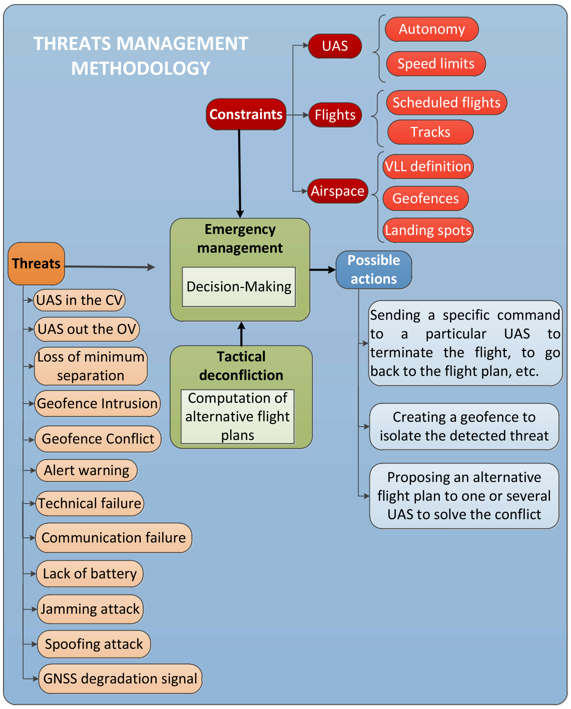


FIGURE 4. Overview of our threat management methodology.

by the operator. These priorities are later used to decide which UAS will modify its flight plan in case of conflict (if two UAS are in conflict, the least critical varies its plan). Then, we consider different threats that may happen during the in-flight phase of the operation, from a predefined set of types. Each type of threat has associated a *severity* level, which will determine the kind of mitigation actions to apply. Given the operations in course and the detected threats, our problem is to decide the best mitigation actions to apply to each UAS, in order to manage all threats and solve the existing conflicts at the same time that the U-space constraints are held.

Figure 4 depicts an overview of the elements considered by our methodology: in orange, the different types of threats; in red, the U-space constraints to be taken into account; in blue, the possible mitigation actions; and in green, the U-space services involved. Once we have a list of detected threats and their types (see details in Section III-B), this information, together with the active U-space constraints, is input to a decision-making procedure that selects the optimal mitigation actions for the required UAS. A multi-objective optimization is carried out in order to select actions maximizing efficiency and safety. The different mitigation actions and the selection procedure will be described in Section III-C. As it was explained in Section II-A, our threat management methodology is applied in real time, using in-flight U-space services. In particular, the decision-making procedure is implemented through the Emergency Management service, which uses the Tactical Deconfliction service to compute alternative flight plans when needed.

B. THREAT TYPES

Reviewing the literature on threat management (see Section II-B), we have identified a list of relevant threats that cover most of the unexpected events that may occur during multi-UAS operations in the U-space. In the following, we describe the different types of threats that we consider:

- *UAS within its Contingency Volume*. The UAS is out of its Flight Geometry but still within its Contingency Volume. In this situation, the UAS is considered under control, because it is still within the Operational Volume, but minor mitigation actions could still be applied so that it returns to its Flight Geometry.
- *UAS out of its Operational Volume*. In this situation, the UAS is considered out of control, as it is flying out of its Operational Volume. Therefore, a mitigation action will be required to solve this occurrence.
- *Loss of minimum separation*. In U-space, there is a minimum safety distance between each pair of UAS, which is determined by the sum of the radii of both Operational Volumes. If two UAS are closer than the safety distance, a mitigation action will be necessary to avoid a potential collision.
- *Geofence intrusion*. This happens when a UAS enters a geofence, i.e., a forbidden 4-dimensional volume (e.g., a static no-fly zone specified by the authorities before operation or a restricted area dynamically created by the U-space during operation). In this situation, a mitigation action to leave the volume will be mandatory.
- *Geofence conflict*. This happens when a UAS detects along its flight plan a geofence that was not planned to be there. In this case, the UAS should avoid entering that geofence and then resume its operation.
- *Alert warning*. Authorities (e.g., fire-fighters, emergency corps, etc.) or stakeholders could notify a wildfire, a bad weather forecast, or any other threatening event. Those occurrences should be managed with the corresponding actions.
- *Technical failure*. A technical failure is an unwanted error of technology-based systems. In the case of UAS, this can involve hardware or software components.
- *Communication failure*. This entails a loss of communication between the UAS and the U-space service provider, which is an event of difficult solution. Emergency actions will be taken in order to mitigate potential damages.
- *Lack of battery*. This event implies the impossibility of ending the UAS operation ordinarily.
- *Jamming attack*. A jamming attack consists of an attempt to jeopardize the *Global Navigation Satellite System* (GNSS) signal of a UAS.
- *Spoofing attack*. A spoofing attack is a situation in which a malicious person or software fakes the UAS information, e.g., so that it seems to be located somewhere else, instead of at its right location. This kind of threat is rather difficult to detect.

TABLE 1. Specified data for each threat.

Threat data type		
Field type	*Field name*	*Description*
integer	threat_type	Threat type, as described in Section III-B
integer	threat_id	Each new threat has a unique identifier
integer[]	uav_ids	List with identifiers of the threatened UAS
integer[]	geofence_ids	List with identifiers of the geofences involved
integer[]	priority_ops	Priorities of the operations involved in the threat
4D waypoint	location	Waypoint (x,y,z,t) where the threat was detected

TABLE 2. U-space constraints.

Constraints		
Source	*Name*	*Description*
UAS	Autonomy	Maximum distance the UAS can still flight
	Speed limit	Maximum UAS speed
Flights	Scheduled flights	Flight plans of all UAS
	Tracks	UAS positions in real time
Airspace	VLL definition	Volume of air below 150 m above ground level
	Geofences	4D volumes (x,y,z,t) for each restricted flight zone
	Landing spots	Waypoints in (x,y) where UAS could land

- *GNSS degradation signal*. In an era of increasing wireless radio frequency congestion, GNSS systems are becoming more at risk of signal degradation due to interference. GNSS signal deterioration typically occurs by signal masking caused by natural (e.g., foliage) and man-made (e.g., buildings) obstructions, ionospheric scintillation, Doppler shift, and antenna effects. This degradation could result in partial or total loss of the UAS tracking.

It is worth noticing that, although we focus on the previous list of threats (some of them are tested in Section IV with field trial use cases), we believe those categories are general enough to accommodate all possible U-space events. Additional events could fit in some of the mentioned types, as they would produce similar effects on UAS flight plans.

C. DECISION-MAKING PROCEDURE

This section describes our decision-making procedure to select the best mitigation actions for each UAS. In terms of U-space architecture [19], this decision-making procedure is implemented within the Emergency Management (EM) service. The EM component is in charge of centralizing all information related to the events that may become a threat, and applying the corresponding mitigation actions. Besides, if an alternative flight plan needs to be computed for the UAS in conflict, the EM relies on the support of the Tactical Deconfliction (TD) service, which is a U-space component providing non-conflicting flight plans.

```
Algorithm 1: Decision-Making Procedure
   Input  : $\mathcal{T} \leftarrow$ list<threat>,
            $\mathcal{C} \leftarrow$ uspace_constraints
1  $\mathcal{S} \leftarrow$ obtain_severities($\mathcal{T}$)
2  $\mathcal{T} \leftarrow$ sort($\mathcal{T}$,$\mathcal{S}$)
3  foreach th in $\mathcal{T}$ do
4      $Type \leftarrow$ action_type(th.threat_type)
5      if $Type == A$ then
6          $a \leftarrow$ newAction(th,$\mathcal{C}$,TYPE_A)
7          sendAction($a$)
8      else if $Type == B$ then
9          $a \leftarrow$ newAction(th,$\mathcal{C}$,TYPE_A)
10         $b \leftarrow$ newAction(th,$\mathcal{C}$,TYPE_B)
11         sendAction($a$,$b$)
12     else if $Type == C$ then
13         $\mathcal{M} \leftarrow$ TD($th$,$\mathcal{C}$)
14         $\xi \leftarrow$ bestManeuvers($\mathcal{M}$)
15         foreach $i$ in $N$ do
16             $c \leftarrow$ newAction(th,$\mathcal{C}$,TYPE_C,$\xi[i]$)
17             sendAction($c$)
18         end
19     end
20 end
```

TABLE 3. Definition of the possible mitigation actions proposed by the methodology.

Mitigation action type	*Description*
A	Sending a specific command or notification to a particular UAS operator, e.g., flight termination, going back to the flight plan, alert warning, etc
B	Creating a geofence to isolate the detected threat
C	Proposing an alternative flight plan to one or several UAS operators for solving a conflict

In the decision-making procedure (see Algorithm 1), the EM takes as input the information of each detected threat, as specified in Table 1, together with the U-space constraints described in Table 2. As output, the EM can decide to take the three different types of mitigation actions defined in Table 3: type A, to send a specific command or notification to a particular UAS operator, e.g., flight termination, going back to the flight plan, alert warning, etc.; type B, to create a geofence to isolate the detected threat; and type C, to propose an alternative flight plan to one or several UAS operators for solving a conflict. In case several threats are simultaneously detected, they are solved in order of decreasing severity (lines 1-2 of Algorithm 1). The severity is defined as the level of damage that a threat can cause in the airspace (e.g., in principle, the damage that a spoofing attack can cause is bigger than that of a UAS which leaves its FG volume). The severity level of each type of threat is manually determined by U-space operators.

Regarding the implementation of the mitigation actions, in the actions of type A, the EM acts just sending a command or notification to the corresponding UAS operator through the U-space communication layer (line 7 of Algorithm 1). In the actions of type B, the EM creates a new geofence that will be stored in a database of the U-space architecture. This database is in charge of storing all updated operational information related to both UAS and geofences. Besides creating the geofence, a command or warning is also sent to the UAS (line 11 of Algorithm 1). In the actions of type C, the EM sends alternative flight plans to the UAS involved in a conflicting situation. The EM asks the TD

TABLE 4. Different maneuvers considered to propose alternative flight plans for a UAS.

Identification	*Description*
1	Route to the destination avoiding a geofence
2	Route back home
3	Route to land in a landing spot
4	Route to the destination getting out of a geofence as soon as possible
5	Route avoiding an aerial vehicle which is too close
6	Route to go back as soon as possible to the FG and resume the flight plan

TABLE 5. Mitigation actions applied for each type of threat, as well as the possible maneuvers.

Threat type	*Mitigation actions applied*	*Possible maneuvers*
UAS in the CV	A	Not applicable
UAS out OV	C	2
	C	3
	C	6
Loss of separation	C	2
	C	3
	C	5
Geofence intrusion	C	2
	C	3
	C	4
Geofence conflict	C	1
	C	2
	C	3
Alert warning	A & B	Not applicable
Technical failure	A & B	Not applicable
Communication failure	B	Not applicable
Lack of battery	C	2
	C	3
Jamming attack	A & B	Not applicable
Spoofing attack	A & B	Not applicable
GNSS degradation signal	C	2
	C	3

module for alternative flight plans, providing information related to the threat to solve (i.e., the type of threatening situation, the data of the affected UAS operations, and the active geofences). Then, the TD attempts different types of maneuvers, selected from those in Table 4, to generate a list of alternative flight solutions for the involved UAS (line 13 of Algorithm 1). Last, the EM chooses the optimal solution for each UAS among the possible alternatives according to a multi-objective optimization problem (line 14 of Algorithm 1). It is important to remark that, although our methodology has the ability to operate autonomously, the current regulatory restrictions do not allow to operate UAS in a totally autonomous manner. Human supervision for accepting or rejecting the alternative flight plans is still mandatory. Nonetheless, we expect more flexibility in the near future in terms of regulation, as authorities are pushing for an Unmanned Aircraft System Traffic Management as much automated as possible.

A threat management methodology should be able to evaluate the operation and propose alternative solutions that are safe, minimizing risks. However, a UAS operation not only needs to be safe but also efficient. This is why we propose a multi-objective optimization to select the best mitigation actions, trading off efficiency and safety. For that, two metrics are defined for each of the possible maneuvers: *cost* and *riskiness*. The former evaluates how costly the maneuver is with respect to the original plan in terms of the additional distance covered (efficiency); the latter indicates the level of risk that the maneuver implies, i.e., how close it comes to other existing flight plans or geofences (safety). Depending on the type of maneuver, there are two ways of computing its cost:

- If the maneuver avoids the threat with an alternative route and then resumes the initial plan, the cost measures the distance (in meters) to be travelled along that additional path.
- On the contrary, if the UAS operation is aborted and the initial flight plan is replaced by a totally new one, e.g, toward a landing spot or back home, the cost measures the distance (in meters) to be travelled along that new flight plan. In order to favor operation completion, we penalize these maneuvers by adding to the cost the length of the uncovered part of the original flight due to the new plan. This means that the earlier the initial flight plan is interrupted, the higher the penalty.

Additionally, the riskiness of the maneuver can be computed by measuring two metrics:

- *Risk I*: This measures the risk due to conflicting situations generated by the maneuver. In particular, we measure the length (in meters) of the new flight plan that is still in conflict. For instance, in a maneuver to get out of a geofence or to go back to a FG, the initial part of the flight plan will still go through the conflicting volume.
- *Risk II*: This measures the risk of getting close to conflicts. In particular, we measure the minimum distance (in meters) of the new flight plan with respect to existing conflicts. For instance, the closer it gets to an existing geofence, the riskier the maneuver.

Table 5 summarizes the types of mitigation actions and maneuvers that are applicable for each threat. In case of *UAS in the CV*, the UAS is just warned (type A action). In case of *Alert warning*, the UAS is warned (type A action), but a geofence is also created around the dangerous situation (type B action). In case of *Technical failure* or *Jamming/spoofing attack*, the UAS is commanded a flight termination (type A action) and a geofence is created around (type B action). In case of *Communication failure* only the geofence is created (type B action), since the UAS could not be notified. For the remaining cases, different types of avoidance maneuvers are applied (type C action).

Given the threat information (Table 1) and the U-space constraints (Table 2), we determine the applicable actions and maneuvers, as indicated in Table 5. The TD generates possible solutions for the applicable maneuvers, with their associated cost and riskiness, so that the threat is avoided and the constraints met. Constraints regarding UAS autonomy/speed and VLL definition are considered to discard some alternative plans which may be unfeasible. Constraints related to scheduled flights, current UAS tracks, and geofences are included as no-fly zones. The known landing spots are used in maneuvers of type 3, so that the TD computes the flight plans to each of them. In case of a maneuver of type 5, i.e., several conflicting UAS avoiding each other, the TD would compute alternative flight plans for the involved UAS, attempting different avoiding directions to generate multiple solutions. Moreover, priorities are considered to only modify flight plans for those UAS with less priority operations. Once all maneuvers for the UAS involved in a given threat have been computed, the EM selects the best option for each UAS by minimizing the following value function:

$$\sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha \cdot c_{ij} + \beta_1 \cdot r_{ij}^{I} - \beta_2 \cdot r_{ij}^{II} \ , \tag{1}$$

where N and M_i represent the number of conflicting UAS for the given threat and the number of available maneuvers for each UAS, respectively; c_{ij} is the cost incurred if UAS i executes maneuver j; r_{ij}^{I} and r_{ij}^{II} are the riskiness I and II associated with maneuver j executed by UAS i; and $\alpha, \beta_1, \beta_2 \in [0, 1]$ are optimization weights. The values of those weights need to be tuned by a human designer. In general, the system should favor safety over efficiency, so a lower penalization for α is expected. Recall that Equation (1) is only used to select maneuvers in actions of type C ($\mathcal{M}$ in line 13 of Algorithm 1 is an $M \times N$ matrix containing cost and riskiness information for the maneuvers of all involved UAS, whereas ξ is a vector with the best maneuver for each UAS). Actions of type A or B are just selected for certain threats (see Table 5).

1) TACTICAL DECONFLICTION

Our methodology for threat management is general enough to work with different implementations of the TD module. Any algorithm able to provide alternative plans for the conflicting UAS using the defined maneuvers could be used. In this work, we used a particular implementation integrated in a U-space architecture [19]. For situations where the flight plans of several UAS in conflict need to be computed (i.e., due to a loss of separation), a geometric approach based on repulsive forces is used to modify the original flight plans [47]. Basically, the algorithm models the UAS trajectories as cords with electrical charges that repel each other, in order to increase their separation. By applying vertical or horizontal separation maneuvers between the involved UAS trajectories in an iterative procedure, several alternative solutions can be generated. The priorities of the conflicting flight plans are also considered, as the algorithm tends not to modify the flight plans of those UAS whose operations present a higher priority in the U-space.

For other situations where a single UAS needs to compute its flight plan avoiding possible threats, e.g, to avoid a geofence, return to its OV, or go to a landing spot, a heuristic path planner based on the well-known A* algorithm is used. Geofences and running flight plans of other UAS are considered no-fly zones by this path planner.

IV. EXPERIMENTAL RESULTS

This section contains experimental results to showcase the capabilities of the proposed methodology for threat management. The objectives of these experiments are twofold: (i) we show the integration of the methodology in a complete U-space architecture [19], with all its functional modules interacting together to accomplish the specified UAS operations; and (ii) we demonstrate our method operating in real time in field experiments, testing its capabilities to solve different types of conflicts in an automated manner. For that purpose, we have defined three use cases (Section IV-A) involving heterogeneous UAS and several types of conflict. The experimental tests were conducted in the ATLAS Test Centre located in Villacarrillo (Jaén, Spain), which offers to the international aerospace community an aerodrome equipped with excellent technological scientific facilities and a segregated airspace, ideal for experimental flights with UAS. The use cases tested are realistic both in terms of the UAS operational parameters and the experimental setup (Section IV-B). All results of the tests are described in Section IV-C.

A. DEFINITION OF THE USE CASES

We define three use cases using heterogeneous UAS to test different maneuverability, namely, multirotor and fixed-wing aircraft. Two of the use cases involve a pair of UAS performing operations with different priorities and the other one just involves a single UAS. In every use case, different unexpected events or threats show up while the UAS are flying and need to be managed by the UTM system.

Figure 5 depicts the initial flight plans for use case 1. Table 6 summarizes the operational parameters. UAS_1 is a multirotor performing an operation for precision agriculture, while UAS_2 is a fixed-wing aircraft that performs a long-range forest surveillance operation. Note that UAS_2 flew above 150 m (VLL airspace). This was done for safety reasons when operating the particular fixed-wing UAS used in the trials. Given the easier maneuverability of UAS_1, the priority of its operation is set lower. The initial flight plans are such that the UAS do not coincide in space and time throughout their operations. However, we forced a delay in the start of the UAS_1 operation, which resulted in a later violation of the minimum safety distance between both UAS. Thus, this use case is used to test how our threat management methodology is able to detect a loss of separation event between the UAS

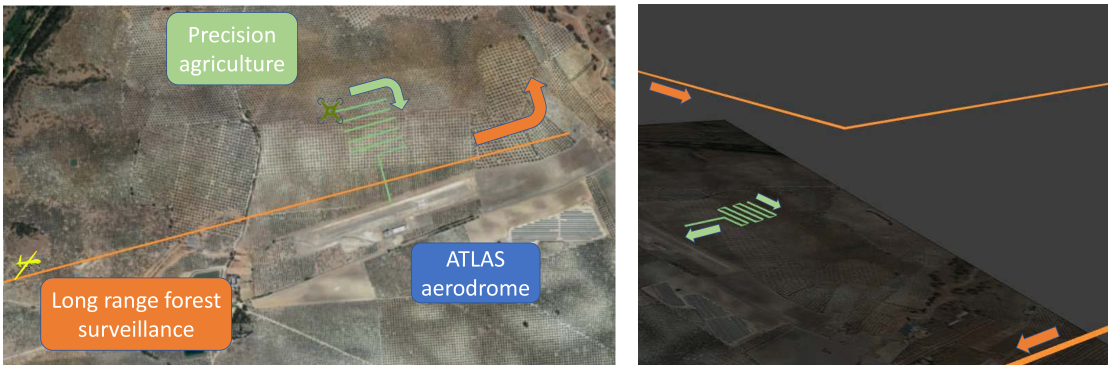


FIGURE 5. Top (left) and perspective (right) views of the initial flight plans for use case 1. All operations were planned in an area of the ATLAS aerodrome in Villacarrillo (Spain).

TABLE 6. Operational parameters for use case 1.

	Operation 1.1	Operation 1.2
ConOps	Precision agriculture	Long-range forest surveillance
UAS type	Multirotor (UAS_1)	Fixed-wing (UAS_2)
Cruising speed	3.3 m/s	30 m/s
Altitude (AGL)	70 m	600 m
Operation priority	Low	High
Threats involved	Loss of separation	Loss of separation

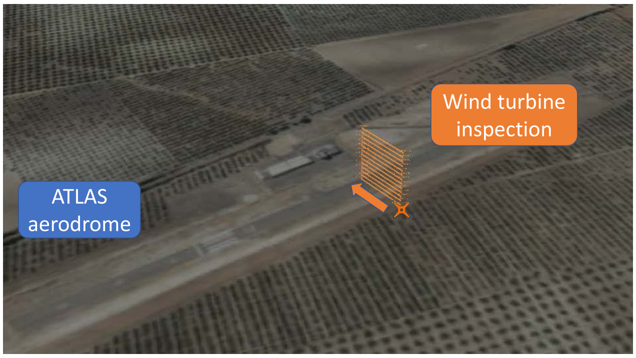


FIGURE 6. Perspective view of the initial flight plan for use case 2. The operation was planned in an area of the ATLAS aerodrome in Villacarrillo (Spain).

and to perform real-time tactical deconfliction for an inter-vehicle conflict, deciding new flight plans for both UAS. Among the available options, the Emergency Management service chooses the optimal solution to solve the conflict.

Figure 6 depicts the initial flight plan for use case 2. Table 7 summarizes the operational parameters. In this case, a multirotor (UAS_3) is used. In its initial flight plan, UAS_3 moves on a vertical sweep to accomplish the inspection of a wind turbine. During the operation, a wildfire notification is simulated close to UAS_3. The objective of this use case is to test how our threat management methodology is able to react in an automated manner to an emergency notified by an external source (e.g.; a wildfire notified by firemen), creating a new geofence (no-fly zone) and then leaving the dangerous area.

TABLE 7. Operational parameters for use case 2.

	Operation 2.1
ConOps	Wind turbine inspection
UAS type	Multirotor (UAS_3)
Cruising speed	3.3 m/s
Altitude (AGL)	30-90 m
Operation priority	Low
Threats involved	Alert warning & Geofence intrusion

TABLE 8. Operational parameters for use case 3.

	Operation 3.1	Operation 3.2
ConOps	Long-range powerline inspection	Event surveillance
UAS type	Fixed-wing (UAS_2)	Multirotor (UAS_3)
Cruising speed	30 m/s	3.3 m/s
Altitude (AGL)	400 m	70-100m
Operation priority	High	Low
Threats involved	Geofence conflict	Jamming attack

Figure 7 depicts the initial flight plans for use case 3 and Table 8 summarizes the operational parameters. UAS_3 is a multirotor performing a surveillance operation, while UAS_2 is a fixed-wing aircraft that has to inspect an electrical powerline. Again, note that UAS_2 flew above 150 m (VLL airspace), due to safety reasons when operating the particular fixed-wing UAS used in the trials. Given the UAS_3 easier maneuverability, the priority of its operation is set lower. The initial flight plans are such that the UAS are not affected by any threat. However, during the operation, we simulated a jamming attack over UAS_3. The objective of this use case is to test how our threat management methodology is able to react in an automated manner to this emergency (jamming attack), creating a new geofence around the UAS attacked and then avoiding to fly inside that geofence.

B. EXPERIMENTAL SETUP

The experimental campaign shown in this paper was carried out within the framework of the GAUSS project. The two UAS depicted in Figure 8 were used, in order to test

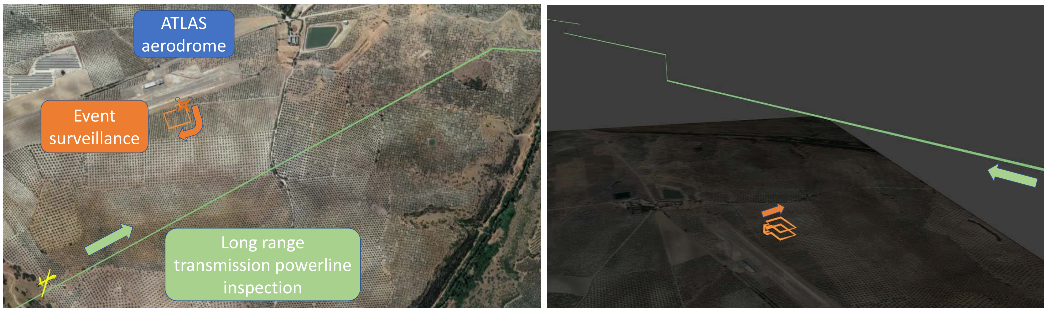


FIGURE 7. Top (left) and perspective (right) views of the initial flight plans for use case 3. All operations were planned in an area of the ATLAS aerodrome in Villacarrillo (Spain).

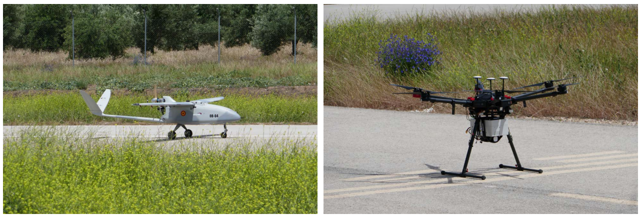


FIGURE 8. The Atlantic I (left) and DJI M600 Pro (right) UAS used in the field experiments.

TABLE 9. Main features of the UAS.

Model	*Type*	*Dimensions*	*MTOM*	*Range*	*Payload*
M-600 Pro (DJI)	Multirotor	*Diameter: 1.5m* *Height: 0.5m*	15kg	500m	3kg
Atlantic I	Fixed-wing	*Wingspan: 3.8m* *Length: 2.8m*	50kg	100km	3.5kg

their heterogeneous maneuverability and different autopilots. Table 9 summarizes the main features of those UAS.

Our threat management methodology is implemented in Python, using the ROS (*Robotic Operating System*) middleware.[5] This methodology is integrated within the complete U-space architecture that was developed in GAUSS, which can be found as open-source code.[6] For software integration and preliminary testing, we used a simulation based on ROS.

The whole experimental setup for the field experiments is shown in Figure 9. A *Ground Control Station* (GCS) was established for each UAS, with proprietary software of the company EVERIS,[7] who provided the aircrafts. This was connected to a *Remote Pilot Station* (RPS) with a Graphical User Interface (*RPS Client Application*) developed by the company SATWAYS.[8] The RPS Client Application, depicted in Figure 10, was in charge of showing telemetry and other operational data to the safety pilot. The computers on board the UAS (Intel NUC) where in charge of producing real-time telemetry data for the operation. A *RPS MQTT Broker* on the RPS was used to communicate data over the Internet to the UTM system, which ran on a different computer on the ground, implementing the U-space services involved in threat management: Emergency Management and Tactical Deconfliction. The UAS RPS communicated with the UTM system exchanging JSON (*JavaScript Object Notation*) messages sent over the MQTT (*Message Queuing Telemetry Transport*) protocol.[9] This hardware setup is realistic in terms of the U-space ecosystem, where the UTM system control station is supposed to be at a different physical location than the UAS operators, communicating via internet. Note that, in case of situations with a large number of UAS sharing the airspace, the methodology would still be scalable, as the decision-making process would just need to take into account local conflicts with nearby UAS. Besides, a cloud-based distributed architecture for the Emergency Management and Tactical Deconfliction modules could be thought.

C. RESULTS

This section presents the results of experimental tests for the three proposed use cases.[10] The main objective is to demonstrate the actual implementation of our methodology in field tests and to assess its feasibility to handle different types of threats in real time and autonomously, only supervised by human safety pilots.

Figure 11 shows a timeline for an experiment implementing use case 1. According to their initial flight plans, both UAS were supposed to start their operations simultaneously at $t = 0\ s$. However, in order to test the system, we simulated a delay of 11 s in the start of the UAS$_1$ operation, which produced a conflict between the two flight plans. Once UAS$_1$ and UAS$_2$ were flying, this conflict, which was a *loss of separation* between both UAS in the last part of their operation, was detected and notified to the EM module ($t = 12.2\ s$). By means of our threat management methodology, the EM module evaluated the type of threat and the priorities of each UAS operation, and it decided to apply a mitigation action of type C. For that, the TD module was asked for support ($t = 12.4\ s$) to attempt different maneuvers, and it computed the flight plans whose metrics are depicted in Table 10.

[5] https://www.ros.org/

[6] https://github.com/grvcTeam/gauss

[7] https://www.everis.com/global/en

[8] https://www.satways.net

[9] We used the open-source Apache Active MQ broker.

[10] An illustrative video with the use cases can be seen at https://www.youtube.com/watch?v=VuoiVHK5-gE

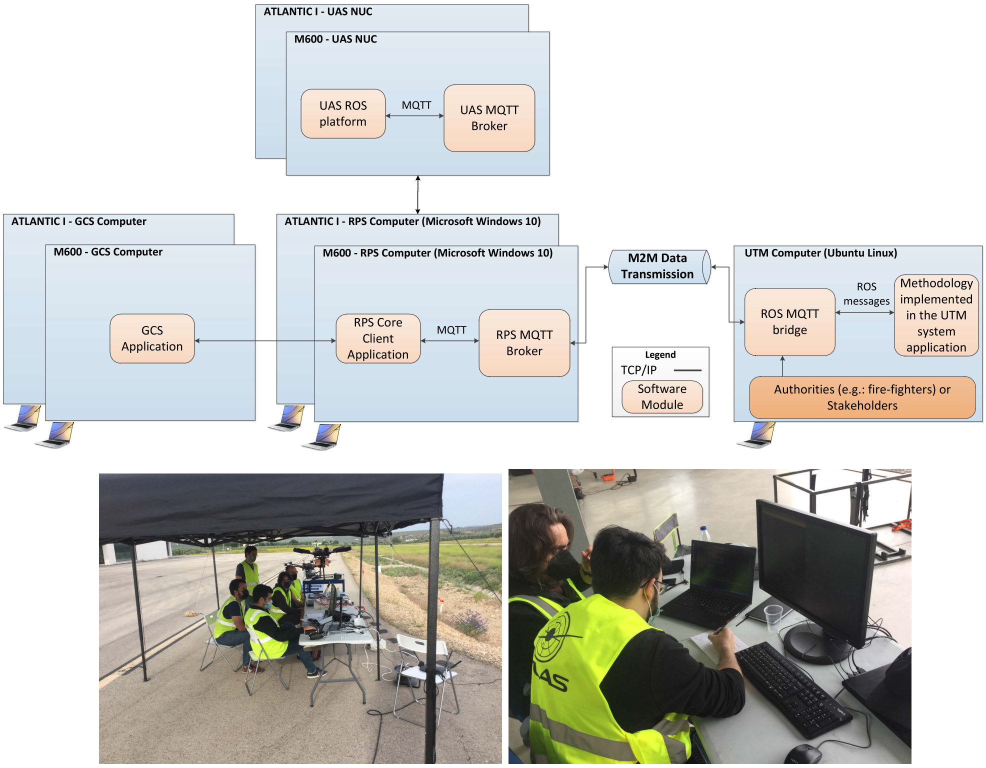

ATLANTIC I - UAS NUC
M600 - UAS NUC
UAS ROS platform
MQTT
UAS MQTT Broker
ATLANTIC I - GCS Computer
M600 - GCS Computer
GCS Application
ATLANTIC I - RPS Computer (Microsoft Windows 10)
M600 - RPS Computer (Microsoft Windows 10)
RPS Core Client Application
MQTT
RPS MQTT Broker
M2M Data Transmission
UTM Computer (Ubuntu Linux)
ROS MQTT bridge
ROS messages
Methodology implemented in the UTM system application
Authorities (e.g.: fire-fighters) or Stakeholders
Legend
TCP/IP
Software Module
UAS

FIGURE 9. Setup for the experiments. On top, an view of the interfaces between the components run on each computer. The computers running the RPS for the two UAS and the UTM system were communicated through the internet via the MQTT protocol. At the bottom, pictures of the UAS Ground Control Station (left) and the UTM computer (right).

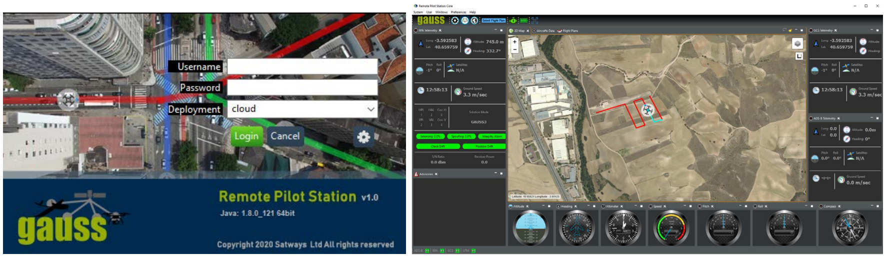

Username
Password
Deployment
cloud
Login
Cancel
gauss
Remote Pilot Station v1.0
Java: 1.8.0_121 64bit
Copyright 2020 Satways Ltd All rights reserved

FIGURE 10. Screenshots of the Graphical User Interface developed by SATWAYS running on the RPS client application.

TABLE 10. Different maneuvers computed by our method for use case 1. The selected solution (in bold) is that with the minimum weighted sum of cost and riskiness.

Maneuver	*Description*	*Cost (m)*	*Riskiness I (m)*	*Riskiness II (m)*	*Total value (m)*
2	Go back home	1523.32	0.00	829.65	295.15
3	Land in landing spot 1	1985.30	0.00	1047.25	480.00
3	Land in landing spot 2	2213.50	0.00	1047.25	571.23
5	Turn right	550.43	0.00	829.65	-28.72
5	Turn left	494.88	0.00	727.87	-20.41
5	**Go down**	**128.52**	**0.00**	**829.67**	**-197.50**

The option to go back home is checked by default, as well as landing on two predefined spots. Recall that these options are penalized adding to the cost the length of the initial flight plan that is not covered. Besides, three different options so that one UAS (the one with the least priority) avoids the other are evaluated. All resulting flight plans are compared in terms of cost and riskiness. The weights were set by design to $\alpha = 0.4$, $\beta_1 = 0.3$ and $\beta_2 = 0.3$, in order to prioritize safety over efficiency. In this use case, the optimal solution was that the multirotor, which had more maneuverability, went down some meters to avoid the conflict and finish its operation, while the fixed-wing UAS kept its flight plan. This solution was notified by the EM to the UAS$_1$ operator ($t = 12.53\ s$). Figure 12 shows the resulting flight plan executed in the field trials.

It is important to highlight that, although our UTM system is able to handle threats in an autonomous fashion, all mitigation actions were sent to the UAS operators for confirmation. This was done for operational safety reasons. Moreover, this is in line with the current U-space regulation, which states that U-space services can only suggest automatically possible correction actions, but those must be accepted or rejected by each UAS operator eventually. Nonetheless, our approach

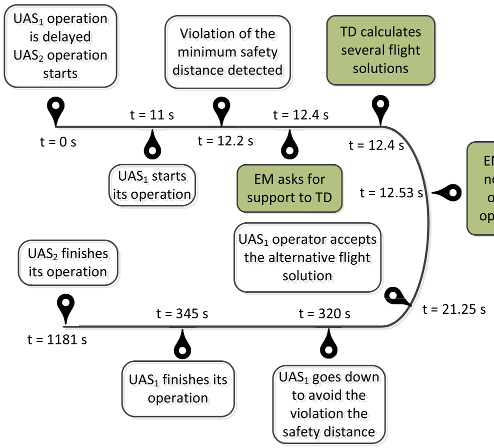

UAS₁ operation is delayed UAS₂ operation starts
Violation of the minimum safety distance detected
TD calculates several flight solutions
t = 0 s
t = 11 s
t = 12.2 s
t = 12.4 s
t = 12.4 s
UAS₁ starts its operation
EM asks for support to TD
EM selects and notifies to UAS operators the optimal solution
t = 12.53 s
UAS₂ finishes its operation
UAS₁ operator accepts the alternative flight solution
t = 21.25 s
t = 1181 s
t = 345 s
t = 320 s
UAS₁ finishes its operation
UAS₁ goes down to avoid the violation the safety distance

FIGURE 11. Timeline of the experiment for use case 1, where a loss of separation is resolved. The events involving the threat management methodology are shown in green.

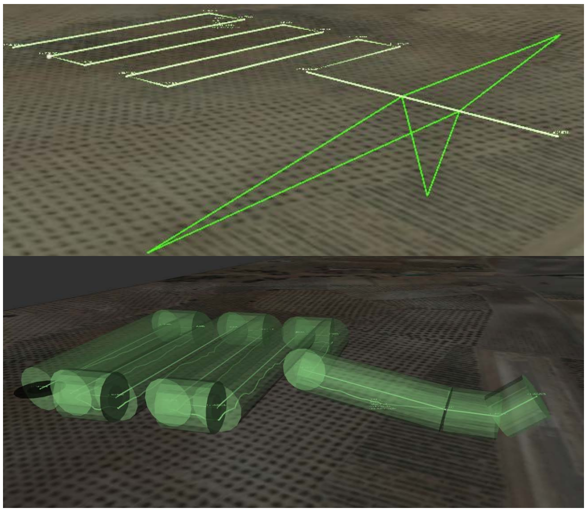


FIGURE 12. Resulting flight plan for UAS_1 in use case 1. On top, the initial flight plan and the alternative solutions to avoid the other UAS (going back home and to the landing spots are not shown for an easier visualization). At the bottom, the selected solution, its Operational Volume, and the actual trajectory followed by UAS_1.

would be able to accommodate threat management based on EM and TD U-space services where the whole process were executed autonomously without the need for human intervention, which is the final objective in the U-space.

Figure 13 shows a timeline for an experiment implementing use case 2. UAS_3 started its operation at $t = 0\ s$, following its initial flight plan. While UAS_3 was flying, a wildfire was notified by the firemen in a nearby location ($t = 2.43\ s$), resulting in a threat of type *alert warning*. Upon that threat, the EM module decided to create a geofence around the fire ($t = 2.63\ s$), to protect aircraft around. Since UAS_3 was within the geofence, a threat of type *geofence intrusion* was detected and notified to the EM module ($t = 3.14\ s$). At that moment, our threat management methodology decided to apply a mitigation action of type C, given the type of threat. For that, the TD module was asked for support ($t = 3.15\ s$) to attempt different maneuvers, and it computed the flight plans whose metrics are depicted in Table 11.

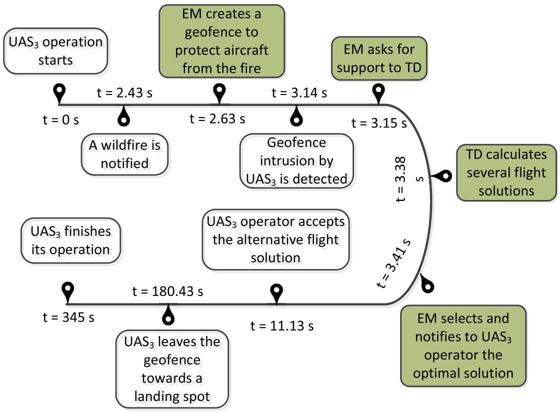

UAS₃ operation starts
EM creates a geofence to protect aircraft from the fire
EM asks for support to TD
t = 0 s
t = 2.43 s
t = 2.63 s
t = 3.14 s
t = 3.15 s
A wildfire is notified
Geofence intrusion by UAS₃ is detected
t = 3.38 s
TD calculates several flight solutions
UAS₃ finishes its operation
UAS₃ operator accepts the alternative flight solution
t = 3.41 s
t = 345 s
t = 180.43 s
t = 11.13 s
UAS₃ leaves the geofence towards a landing spot
EM selects and notifies to UAS₃ operator the optimal solution

FIGURE 13. Timeline of the experiment for use case 2, where an alert warning (wildfire notification) and a geofence intrusion are resolved. The events involving the threat management methodology are shown in green.

TABLE 11. Different maneuvers computed by our method for use case 2. The selected solution (in bold) is that with the minimum weighted sum of cost and riskiness.

Maneuver	*Description*	*Cost (m)*	*Riskiness I (m)*	*Riskiness II (m)*	*Total value (m)*
3	**Land on landing spot 1**	**197.97**	**94.00**	**0.00**	**107.39**
3	Land on landing spot 2	382.55	122.00	0.00	189.62

TABLE 12. Different maneuvers computed by our method for use case 3. The selected solution (in bold) is that with the minimum weighted sum of cost and riskiness.

Maneuver	*Description*	*Cost (m)*	*Riskiness I (m)*	*Riskiness II (m)*	*Total value (m)*
1	**Route avoiding the geofence**	**1400.29**	**0.00**	**214.84**	**495.92**
2	Go back home	12393.85	0.00	9002.26	2256.94
3	Land on landing spot 1	20943.95	0.00	469.57	8237.00
3	Land on landing spot 2	20855.74	0.00	622.65	8155.50

Options to go back home and to get out of the geofence as soon as possible and resume the flight plan were discarded, as they did not fulfil the U-space constraints. This happened because the whole initial flight plan of UAS_3 was within the created geofence. Alternative options to land on the known landing spots were checked instead. All resulting flight plans were compared in terms of cost and riskiness. The weights were also set by design to $\alpha = 0.4$, $\beta_1 = 0.3$ and $\beta_2 = 0.3$. In this use case, the optimal solution was that the multirotor landed on the closest landing spot outside the geofence. This solution was notified by the EM to the UAS_3 operator ($t = 3.41\ s$). Figure 14 shows the resulting flight plan executed in the field trials.

Figure 15 shows a timeline for an experiment implementing use case 3. Both UAS started their operations simultaneously ($t = 0\ s$), following their initial flight plans. During their operation, we simulated a jamming attack over UAS_3 ($t = 12\ s$). For that type of threat, our method for threat management created a geofence around the attacked UAS (action of type B) and asked the UAS_3 operator (action of type A) to land now ($t = 12.1\ s$). While UAS_3 was landing, a geofence conflict between UAS_2 and the new geofence was detected and notified ($t = 12.38\ s$), i.e., the flight plan of UAS_2 was going through that new geofence. Our threat management methodology decided to apply a mitigation action of type C, given the type of threat. For that,

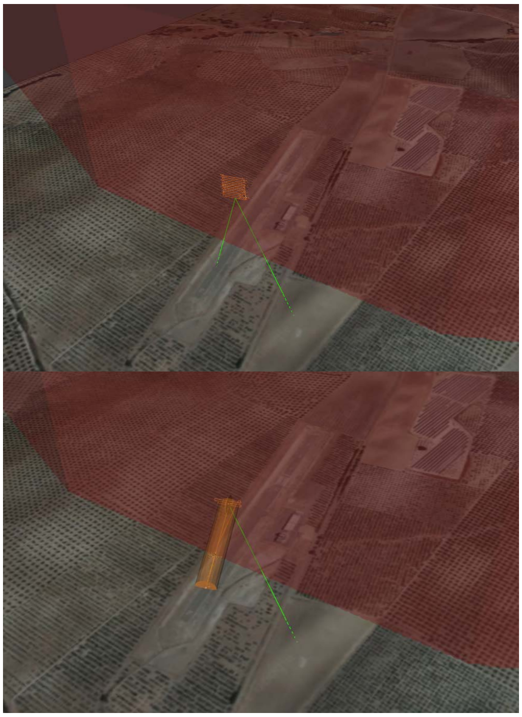


FIGURE 14. Resulting flight plan for UAS_3 in use case 2. On top, the initial flight plan and the alternative solutions to land on different spots. At the bottom, the selected solution to the closest landing spot, its Operational Volume, and the actual trajectory followed by UAS_3. In red, the created geofence.

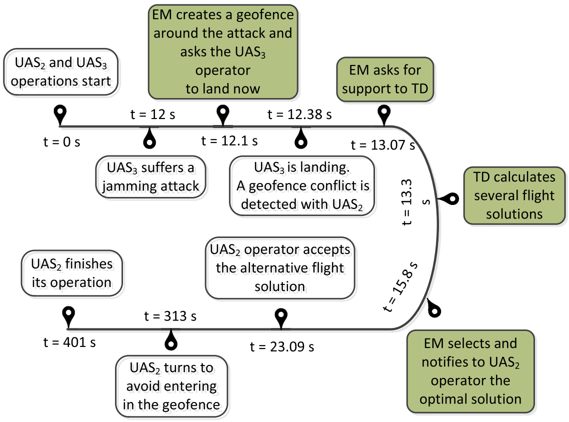

UAS_2 and UAS_3 operations start
EM creates a geofence around the attack and asks the UAS_3 operator to land now
EM asks for support to TD
t = 12 s
t = 12.38 s
t = 0 s
t = 12.1 s
t = 13.07 s
UAS_3 suffers a jamming attack
UAS_3 is landing. A geofence conflict is detected with UAS_2
t = 13.3 s
TD calculates several flight solutions
UAS_2 finishes its operation
UAS_2 operator accepts the alternative flight solution
t = 15.8 s
t = 313 s
t = 401 s
t = 23.09 s
EM selects and notifies to UAS_2 operator the optimal solution
UAS_2 turns to avoid entering in the geofence

FIGURE 15. Timeline of the experiment for use case 3, where a jamming attack and a geofence conflict are resolved. The events involving the threat management methodology are shown in green.

the TD module was asked for support ($t = 13.07\ s$) to attempt different maneuvers, and it computed the flight plans whose metrics are depicted in Table 12.

Options to land on the known landing spots and to go back home are checked by default. Besides, an additional option so that UAS_2 avoids the geofence and resumes its flight plan. All resulting flight plans are compared in terms of cost and riskiness. The weights were also set by design to $\alpha = 0.4$, $\beta_1 = 0.3$ and $\beta_2 = 0.3$. In this use case, the optimal solution was that the fixed-wing UAS circumvented the geofence and then resumed with its original plan. This solution was notified by the EM to the UAS_2 operator ($t = 15.8\ s$). Figure 16 shows the resulting flight plan executed in the field trials.

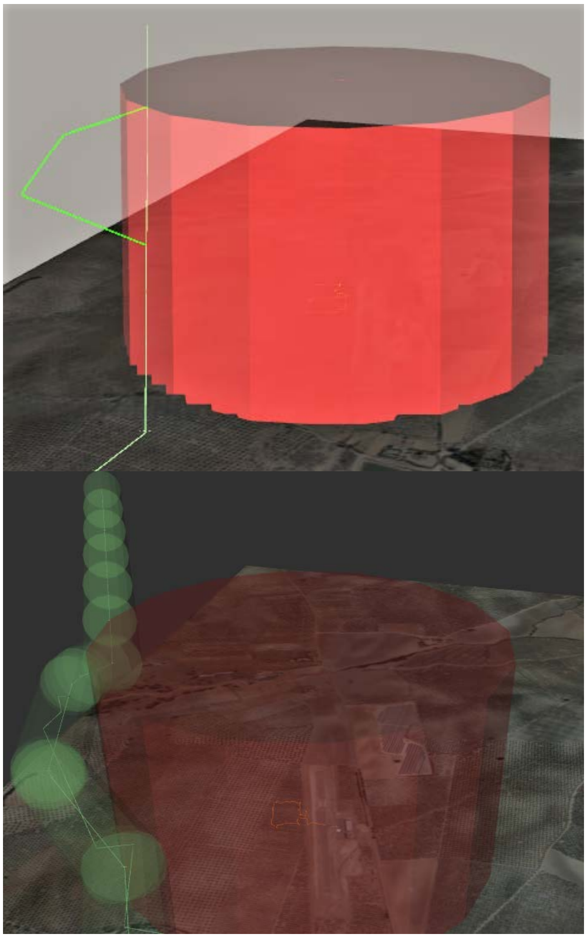


FIGURE 16. Resulting flight plan for UAS_3 in use case 3 (going back home and to the landing spots are not shown for an easier visualization). On top, the initial flight plan and the alternative solution to avoid the geofence (red cylinder). At the bottom, the selected solution going around the geofence, its Operational Volume, and the actual trajectory followed by UAS_3.

V. CONCLUSION

In this paper, we have presented a threat management methodology for UAS operating within the U-space ecosystem. Our method is capable of handling all usual threats in UTM systems, and it performs real-time and autonomous decision-making to provide optimal mitigation actions in terms of cost and risk level. The methodology is integrated with a U-space architecture, implementing in-flight services for emergency management and tactical deconfliction.

We have demonstrated that our methodology is capable of autonomously handling heterogeneous threats in real time, through a set of use cases implemented on real rotary- and fixed-wing UAS. In our experiments, the system was able to resolve different types of conflicts, reasoning about 4D UAS trajectories, geofences, and Operational Volume. Moreover, the experimental setup was realistic with respect to the actual U-space ecosystem, as the onboard and on-ground systems were running at different places and communicated over the Internet.

However, our system has still some limitations. It relies on an accurate positioning of UAS, dismissing possible uncertainties. For instance, during the experiments performed, we noticed that the telemetry of the UAS, especially the fixed-wing aircraft, were unstable at some periods, which could result in the detection of ''fake'' conflicts. These uncertainties could be increased due to communication delays or blackouts. As future work, we plan to introduce security margins in our method to consider uncertainties in the detection and resolution of threats. Besides, this paper could be the base for the design of a digital and automated methodology for risk assessment, working in real time as different UAS are flying and unexpected events show up. The method could be integrated in a real UTM system through the specific U-space service *Risk analysis assistance*.

REFERENCES

[1] (2016). SESAR. *European Drones Outlook Study*. [Online]. Available: https://op.europa.eu/s/oL7Y

[2] J.-P. Aurambout, K. Gkoumas, and B. Ciuffo, ''Last mile delivery by drones: An estimation of viable market potential and access to citizens across European cities,'' *Eur. Transp. Res. Rev.*, vol. 11, no. 1, pp. 1–21, Dec. 2019.

[3] J. Capitan, L. Merino, and A. Ollero, ''Cooperative decision-making under uncertainties for multi-target surveillance with multiples UAVs,'' *J. Intell. Robotic Syst.*, vol. 84, nos. 1–4, pp. 371–386, Dec. 2016.

[4] D. Benjumea, A. Alcántara, A. Ramos, A. Torres-Gonzalez, P. Sánchez-Cuevas, J. Capitan, G. Heredia, and A. Ollero, ''Localization system for lightweight unmanned aerial vehicles in inspection tasks,'' *Sensors*, vol. 21, no. 17, p. 5937, Sep. 2021.

[5] P. Garcia-Aunon, J. J. Roldán, and A. Barrientos, ''Monitoring traffic in future cities with aerial swarms: Developing and optimizing a behavior-based surveillance algorithm,'' *Cognit. Syst. Res.*, vol. 54, pp. 273–286, May 2019.

[6] A. Alcantara, J. Capitan, A. Torres-Gonzalez, R. Cunha, and A. Ollero, ''Autonomous execution of cinematographic shots with multiple drones,'' *IEEE Access*, vol. 8, pp. 201300–201316, 2020.

[7] V. Kramar, ''UAS (drone) in response to coronavirus,'' in *Proc. 27th Conf. Open Innov. Assoc. (FRUCT)*, Sep. 2020, pp. 90–100.

[8] N. Peinecke and A. Kuenz, ''Deconflicting the urban drone airspace,'' in *Proc. IEEE/AIAA 36th Digit. Avionics Syst. Conf. (DASC)*, Sep. 2017, pp. 1–6.

[9] P. Kopardekar. (2015). *Unmanned Aerial System (UAS) Traffic Management (UTM): Enabling Civilian Low-Altitude Airspace and Unmanned Aerial System Operations*. NASA. [Online]. Available: https://n9.cl/6yxi

[10] P. Kopardekar, J. Rios, T. Prevot, M. Johnson, J. Jung, and J. E. Robinson, ''Unmanned aircraft system traffic management (UTM) concept of operations,'' in *Proc. AIAA Aviation Technol., Integr., Oper. Conf.*, 2016, pp. 1–16.

[11] (2017). SESAR. *U-Space Blueprint*. [Online]. Available: https://acortar.link/ATDkWL

[12] (2019). European Commission. *Unmanned Aircraft Systems and on Third-Country Operators of Unmanned Aircraft Systems*. [Online]. Available: https://eur-lex.europa.eu/eli/reg_del/2019/945/oj

[13] (2019). European Commission. *Rules and Procedures for the Operation of Unmanned Aircraft*. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=pi_com: C(2019)3824

[14] C. Capitan, J. Capitan, A. R. Casta no, and A. Ollero, ''Risk assessment based on SORA methodology for a UAS media production application,'' in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2019, pp. 451–459.

[15] (2021). European Commission. *Regulatory Framework for the U-Space*. https://ec.europa.eu/transport/sites/transport/files/c20212671-u-space. %pdf

[16] (2021). European Commission. *Requirements for Providers of Air Traffic Management/Air Navigation Services and Other Air Traffic Management Network Functions in the U-Space Airspace Designated in Controlled Airspace*. [Online]. Available: https://ec.europa.eu/transport/sites/transport/files/c20212672-u-space.%pdf

[17] (2021). European Commission. *Requirements for Manned Aviation Operating in U-Space Airspace*. [Online]. Available: https://ec.europa.eu/transport/sites/transport/files/c20212673-u-space.%pdf

[18] (2019). JARUS. *Joint Authorities for Rulemaking of Unmanned Systems JARUS guidelines on SORA 2.0*. [Online]. Available: http://jarus-rpas.org/sites/jarus-rpas.org/files/jar_doc_06_jarus_sora_% v2.0.pdf

[19] C. Capitán, H. Pérez-León, J. Capitán, Á. Castaño, and A. Ollero, ''Unmanned aerial traffic management system architecture for U-space in-flight services,'' *Appl. Sci.*, vol. 11, no. 9, p. 3995, Apr. 2021.

[20] C. Barrado, M. Boyero, L. Brucculeri, G. Ferrara, A. Hately, P. Hullah, D. Martin-marrero, E. Pastor, A. P. Rushton, and A. Volkert, ''U-space concept of operations: A key enabler for opening airspace to emerging low-altitude operations,'' *Aerospace*, vol. 7, no. 3, pp. 1–18, 2020.

[21] (2020). Eurocontrol. *U-Space Services Implementation Monitoring Report*. [Online]. Available: https://www.eurocontrol.int/publication/u-space-services-implementation%-monitoring-report

[22] European Aviation Safety Agency (EASA). *UAS Safety Risk Portfolio and Analysis*. Tech. Accessed: Jul. 4, 2022. [Online]. Available: https://www.easa.europa.eu/document-library/general-publications/uas-safety-risk-portfolio-and-analysis

[23] (2003). B. Considine. *Guidelines for Controller Training in the Handling of Unusual/Emergency Situations Guidelines for Controller Training in the Handling of Unusual/Emergency Situations*. [Online]. Available: https://skybrary.aero/sites/default/files/bookshelf/15.pdf

[24] M. Finke. (2016). *Defining RPAS Emergency Procedures for Controllers, Remote Pilots and Automatic on-Board Systems*. [Online]. Available: https://bit.ly/3ee1jrR

[25] (2019). CORUS. *Intermediate ConOps Annex E—List of Threats and Events*. [Online]. Available: https://www.sesarju.eu/sites/default/files/documents/u-space/CORUS_Inte% rmediate_ConOps.pdf

[26] (2017). East Gippsland Shire Council. *Emergency Procedures—Drone (Remote Piloted Aircraft)*. [Online]. Available: https://bit.ly/3qkMzNg

[27] L. Fern, R. C. Rorie, and R. Shively, ''UAS contingency management: The effect of different procedures on ATC in civil airspace operations,'' in *Proc. 14th AIAA Aviation Technol., Integr., Oper. Conf.*, Jun. 2014.

[28] E. Pastor, P. Royo, E. Santamaria, X. Prats, and C. Barrado, ''In-flight contingency management for unmanned aerial vehicles,'' *J. Aerosp. Comput., Inf., Commun.*, vol. 9, no. 4, pp. 144–160, Dec. 2012.

[29] M. A. Masri, S. Dbeis, and M. Al Saba, ''Autolanding a power-off UAV using on-line optimization and slip maneuvers,'' *J. Intell. Robotic Syst.*, vol. 86, no. 2, pp. 255–276, May 2017.

[30] L. Mejias and P. Eng, ''Controlled emergency landing of an unpowered unmanned aerial system,'' *J. Intell. Robotic Syst.*, vol. 70, nos. 1–4, pp. 421–435, Apr. 2013.

[31] X. Guo, S. Denman, C. Fookes, L. Mejias, and S. Sridharan, ''Automatic UAV forced landing site detection using machine learning,'' in *Proc. Int. Conf. Digit. Image Computing: Techn. Appl. (DICTA)*, Nov. 2014, pp. 1–7.

[32] M. Warren, L. Mejias, X. Yang, B. Arain, F. Gonzalez, and B. Upcroft, ''Enabling aircraft emergency landings using active visual site detection,'' in *Field and Service Robotics* (Springer Tracts in Advanced Robotics), vol. 105, L. Mejias, P. Corke, and J. Roberts, Eds. Cham, Switzerland: Springer, 2015, doi: 10.1007/978-3-319-07488-7_12.

[33] M. Warren, L. Mejias, J. Kok, X. Yang, F. Gonzalez, and B. Upcroft, ''An automated emergency landing system for fixed-wing aircraft: Planning and control,'' *J. Field Robot.*, vol. 32, no. 8, pp. 1114–1140, Dec. 2015.

[34] A. J. T. Harmsel, I. J. Olson, and E. M. Atkins, ''Emergency flight planning for an energy-constrained multicopter,'' *J. Intell. Robotic Syst.*, vol. 85, no. 1, pp. 145–165, Jan. 2017.

[35] E. M. Atkins, I. A. Portillo, and M. J. Strube, ''Emergency flight planning applied to total loss of thrust,'' *J. Aircr.*, vol. 43, no. 4, pp. 1205–1216, Jul. 2006.

[36] F. Ho, R. Geraldes, A. Goncalves, B. Rigault, B. Sportich, D. Kubo, M. Cavazza, and H. Prendinger, ''Decentralized multi-agent path finding for UAV traffic management,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 997–1008, Feb. 2022.

[37] M. Doole, J. Ellerbroek, and J. M. Hoekstra, ''Investigation of merge assist policies to improve safety of drone traffic in a constrained urban airspace,'' *Aerospace*, vol. 9, no. 3, p. 120, Feb. 2022. [Online]. Available: https://www.mdpi.com/2226-4310/9/3/120

[38] Y. Tang, Y. Xu, and G. Inalhan, ''An integrated approach for on-demand dynamic capacity management service in U-space,'' *IEEE Trans. Aerosp. Electron. Syst.*, early access, Mar. 15, 2022, doi: 10.1109/TAES.2022.3159317.

[39] A. Mcfadyen and L. Mejias, "A survey of autonomous vision-based see and avoid for unmanned aircraft systems," *Progr. Aerosp. Sci.*, vol. 80, pp. 1–17, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0376042115300208

[40] J. Alonso-Mora, T. Naegeli, R. Siegwart, and P. Beardsley, "Collision avoidance for aerial vehicles in multi-agent scenarios," *Auto. Robots*, vol. 39, no. 1, pp. 101–121, 2015.

[41] T. Dubot and A. Joulia, "Towards U-space conflict management services based on 4D protection bubbles," in *Proc. AIAA Aviation Forum*, 2021.

[42] J. Jover, A. Bermúdez, and R. Casado, "A tactical conflict resolution proposal for U-space Zu airspace volumes," *Sensors*, vol. 21, no. 16, p. 5649, Aug. 2021.

[43] A. Jover, J. Bermúdez, and R. Casado, "Priority-aware conflict resolution for U-space," *Electronics*, vol. 11, no. 8, p. 1225, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/8/1225

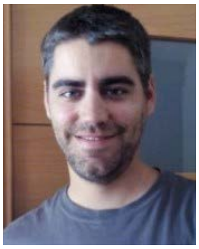
[44] M. Campusano, K. Jensen, and U. P. Schultz, "Towards a service-oriented U-space architecture for autonomous drone operations," in *Proc. IEEE/ACM 3rd Int. Workshop Robot. Softw. Eng. (RoSE)*, Jun. 2021, pp. 63–66.

[45] V. Lappas, G. Zoumponos, V. Kostopoulos, H. I. Lee, H.-S. Shin, A. Tsourdos, M. Tantardini, D. Shomko, J. Munoz, E. Amoratis, A. Maragkakis, T. Machairas, and A. Trifas, "EuroDRONE, a European unmanned traffic management testbed for U-space," *Drones*, vol. 6, no. 2, p. 53, Feb. 2022.

[46] V. Alarcón, M. García, F. Alarcón, A. Viguria, Á. Martínez, D. Janisch, J. J. Acevedo, I. Maza, and A. Ollero, "Procedures for the integration of drones into the airspace based on U-space services," *Aerospace*, vol. 7, no. 9, p. 128, Sep. 2020.

[47] J. J. Acevedo, C. Capitán, J. Capitán, A. R. Castaño, and A. Ollero, "A geometrical approach based on 4D grids for conflict management of multiple UAVs operating in U-space," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Sep. 2020, pp. 263–270.

CARLOS CAPITÁN received the master's degree in electrical energy systems and the master's degree in renewable energies. He was worked with the Power Electronics Group, University of Seville, Spain, from 2011 to 2018, participating in more than 20 European and national research and development projects. Since then, he has been working with the GRVC Robotics Laboratory, on his Ph.D. related with UTM, UAS, U-space, risk analysis, and decision-making methods. He is currently a Senior Innovation Engineer working with the GRVC Robotics Laboratory, University of Seville. He is also an Industrial Engineer with a wide experience in EU project management.



JESÚS CAPITÁN received the Ph.D. degree in telecommunication engineering from the University of Seville, in 2011. He has also worked as a Postdoctoral Researcher with the Instituto Superior Tecnico, Lisbon, Portugal, and the University of Duisburg-Essen, Essen, Germany. During his Ph.D. degree, he was a Visiting Researcher with the Robotics Institute, Carnegie Mellon University, Pittsburgh, USA, and the Instituto Superior Tecnico. He is currently an Associate Professor with the University of Seville. He has participated in more than 25 projects (14 international). He was one of the PI with the University of Seville in the EU project MULTIDRONE and participant of the USE Team in the MBZIRC competition (2017, 2020—winners of the Third Challenge). He is the author of more than 50 publications focusing on multi-robot cooperation, decision making, and multi-UAV conflict resolution. He is an Associate Editor of the International Conference on Robotics and Automation (ICRA), the IEEE Robotics and Automation Letters, and the *International Journal of Advanced Robotics Systems*.

ÁNGEL R. CASTAÑO received the M.Sc. degree in telecommunications engineering and the Ph.D. degree in systems engineering and automation. He was a Visiting Researcher with the Center for Self-Organizing and Intelligent Systems, Utah State University, and the E. Piaggio Center, University of Pisa. He is currently an Associate Professor with the University of Seville. He has participated in more than 30 research projects, including 11 projects funded by the European Commission related to unmanned vehicles and cooperating multi-vehicle systems. He is the PI of the University of Seville in the EU project GAUSS. He was the Team Leader of the Iberian Robotics Team in the MBZIRC 2020 competition (winners of the Third Challenge). He has also led 11 technology transfer projects with several companies, such as Airbus Military, Navantia, and Iturri Group, or EMASESA. His research interests include multi-robot systems and intelligent transportation systems.

ANÍBAL OLLERO (Fellow, IEEE) has been a Full Professor with the Universities of Santiago and Universities of Malaga, Spain, and a Visiting Researcher with the Robotics Institute of Carnegie Mellon University, Pittsburgh, USA, and LAAS-CNRS, Toulouse, France. He is currently a Full Professor, the Head of the GRVC Robotics Laboratory, University of Seville, and the Scientific Advisor of the Center for Advanced Aerospace Technologies, Seville, Spain. He has authored more than 750 publications, including nine books, 200 journal articles, and 15 books edited. He has delivered plenaries and keynotes in more than 100 events, including IEEE ICRA 2016 and IEEE IROS 2018. He has been a Supervisor of 45 Ph.D. theses and led more than 160 research projects, participating in more than 25 projects of the European Research Program (being the Coordinator of seven and an Associated Coordinator of three), all of them dealing with unmanned aerial systems and aerial robots. Since November 2018, he has been running the GRIFFIN ERC-Advanced Grant, and since December 2019, he has been coordinating the H2020-AERIAL-CORE Project about aerial robotic manipulators for inspection and maintenance. He has transferred technology to 20 companies and has been awarded with 23 international research and innovation awards, including the Overall Information and Communication Technologies Innovation Radar Prize 2017 of the European Commission and the recent Rey Jaume I in New Technologies (Spain). He has also been elected between the three European innovators of the year, being candidate to the European personalities, in 2017 and IEEE Fellow "for contributions to the development and deployment of aerial robots." He was a member of the "Board of Directors" of euRobotics, until March 2019. He was also the Founder and the President of the Spanish Society for the Research and Development in Robotics (SEIDROB), until November 2017. He is currently the Co-Chair of the IEEE Technical Committee on Aerial Robotics and Unmanned Aerial Vehicles, and the Coordinator of the Aerial Robotics Topic Group, euRobotics.