Intelligent Information Management, 2021, 13, 97-123
https://www.scirp.org/journal/iim
ISSN Online: 2160-5920
ISSN Print: 2160-5912

Unmanned Aerial Vehicles Flight Safety Improvement Using In-Flight Awareness

André L. P. Mattei[1], Engenharia S. A. Orbital[1], Claudio F. M. Toledo[2], Jesimar da Silva Arantes[2], Onofre Trindade Jr.[2]

[1]São José dos Campos, SP, Brazil
[2]Department of Computer Systems, Institute of Mathematics and Computer Sciences, University of São Paulo, Sao Carlos, SP, Brazil
Email: mattei@orbitalengenharia.com.br, claudio@icmc.usp.br, otj@icmc.usp.br, jesimar.arantes@gmail.com

How to cite this paper: Mattei, A.L.P., Orbital, E.S.A., Toledo, C.F.M., da Silva Arantes, J. and Trindade Jr., O. (2021) Unmanned Aerial Vehicles Flight Safety Improvement Using In-Flight Awareness. *Intelligent Information Management*, **13**, 97-123.
https://doi.org/10.4236/iim.2021.132005

Received: December 30, 2020
Accepted: March 5, 2021
Published: March 8, 2021

Abstract

This paper presents a novel onboard system called In-Flight Awareness Augmentation System (IFA^{2}S) to improve flight safety. IFA^{2}S is designed to semi-automatically (with human supervision) avoid hazards and accidents due to either internal or external causal factors. The requirements were defined in an innovative way using Systems-Theoretic Process Analysis (STPA) method and applied next to model the system. IFA^{2}S increases aircraft awareness regarding both itself and its environment and, at the same time, recognizes platform and operational constraints to act in accordance to predefined decision algorithms. Results are presented through simulations and flight tests using state machines designed to allow the adoption of appropriate actions for the identified hazards. The different decision algorithms are evaluated over as many as possible hazard situations by simulations conducted with software Labview and XPlane flight simulator. Flight tests are performed in a small fixed wing aircraft and make use of a limited version IFA^{2}S, partially attending identified requirements. Results support the conclusion that IFA^{2}S is capable of improving flight safety.

Keywords

Unmanned Aerial Vehicles, Air Safety, Systems-Theoretic Process Analysis, In-Flight Awareness

1. Introduction

Unmanned Aerial Vehicles (UAVs) are becoming more robust in terms of processing power, autopilot (AP), embedded sensors and flight time, which leverage the use of such aerial robots in real world applications for agriculture,

transportation, logistics and surveillance scenarios. However, the current level of autonomy and decision making available in the UAV can be enhanced by employing computer systems that lead the level of autonomy for UAVs changes from a ground control system, with a human pilot in charge, to a fully autonomous flight.

The development of systems for UAV, focused on autonomous decisions capabilities, is challenging since it is expected a chance of failure less than the accepted for general aviation. The high accident rates presented by Unmanned Aerial Vehicles (UAV) have given rise to debate about the risks involved in their operation. It is recognized that the UAV operators shall be aware of the external and internal conditions (surrounding environment and health system, respectively). The complexity and the number of technical and subjective factors involved in the control of an UAV create conditions where the pilot does not act in a timely manner or does it wrongly.

This paper approaches this matter giving more autonomy and perceptions to the aircraft via both a new onboard system named In-Flight Awareness Augmentation System (IFA^{2}S) and its reference model In-Flight Awareness (IFA). The purpose of this work differs from other literature approaches since it emphasizes the concept of Situational Awareness (SA). The objective of this paper is to propose a novel autonomous decision-taker onboard the aircraft to improve flight safety. The general idea is to make the UAV more conscious (situational awareness increase) about its subsystems conditions (internal health), flight profile, intruders presence (other aircrafts), and surrounding conditions (ground and meteorological), keeping pilots on the ground as system managers.

This paper is organized as follows: Section 2 presents a literature review regarding aspects related to this work and Section 3 begins the development of IFA^{2}S from the definition of its requirements using the STPA method. IFA concepts and the methodology used to model IFA^{2}S are described in Section 4. Section 5 reports simulations where the IFA2S model is built using Labview software and stressed under some critical situations previously identified using XPlane flight simulator. Section 6 presents flight tests results using an IFA^{2}S system developed for a small fixed wing aircraft. The conclusions follow in Section 7.

2. Related Work

Nowadays, the technologies embedded on UAVs are more robust in terms of processing power, autopilot (AP), embedded sensors and flight time, which have leveraged the application of such aircraft to many real world scenarios in agriculture, transportation, logistics, surveillance, among others [1] [2] [3] [4]. The wide area of applications imposes to address the chance of failure for UAVs systems that must be less than the accepted for general aviation [5] [6]. Nevertheless, since the pilots are not inside the aircraft to have their sensorial means available, only the data presented in a display are at hand [7] [8]. As highlighted

by [1] and [9], the flexibility provided by the autonomy increase exposes the platform to degradation in the system performance due to environmental variability and distributed decision making (human x electronics). Thus, a platform-centric SA is proposed, instead of relying on human pilots' perceptions as in [10]. The paper also innovates since IFA^{2}S design is completely based on Systems-Theoretic Process Analysis (STPA) method [11] [12], aiming to allow the system to act as soon as it identifies a situation that potentially leads to an accident.

This paper describes a process similar to that found in articles [13] and [14] to achieve SA in dynamic systems. At level 1, data from internal and external sources are collected; in level 2, an algorithm uses this data to define the current situation; and in level 3, the system acts accordingly to achieve a desired situation, the concept of Situational Awareness (SA), given by [10]: "*the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future*". Information is a key factor on SA and most of the literature considers the operator's perception about the current situation. The authors in [15] consider elements such as the status of the aircraft systems, climate conditions, the payload condition, and knowledge of the operator on both the platform capabilities and dynamic aspects involved (level 1). The importance of data processing is highlighted in [16] [17] and it may come from different sources and emphasize a proper decision-making process to manage successfully crises situations (level 2). The authors in (1) state that operators' SA depends on data availability and their understanding based on the context in order to design actions in the future for a semi-autonomous mode (level 3). In this work, IFA^{2}S controls aircraft under pilot's supervision.

Being a main concern in aviation, IFA also covers aspects related to avoiding air collisions, generally called Sense and Avoidance (S&A). Although [18] considers the pilot on the ground as responsible for both detecting and keeping safe distances from other aircraft, many authors evaluate ways for the platform to prevent this kind of accident autonomously. A number of different approaches are possible: passive as in [19]; active as in [20]; using data links as [21]; and using ADS-B as in [22].

Besides surrounding air traffic, IFA also avoids ground proximity and fly over certain areas, such as populated and sensitive (military, nuclear etc.). Some situations during the flight may obligate the aircraft to change the route previously planned. Most of the literature considers only external sources for designing a new route, such as [23] and [24]. In addition to external sources, IFA also considers that some internal resources can be used. For example, critical conditions may obligate an emergency landing, such as an imminent failure in an internal system, where an internal re-planning algorithm can define an emergency landing route as proposed in [25] [26].

Systems-Theoretic Accident Model and Processes (STAMP) is defined by [27]

as "*a new type of accident model based on systems theory rather than the traditional analytic reduction and reliability theory*". STPA is a hazard analysis technique based on STAMP and defined by [27] as "*a new hazard analysis technique based on systems thinking and a new model of accident causation based on systems theory rather than reliability theory*". STPA general description and usefulness in designing safer complex software-intensive systems may be found in [28]. As demonstrated by [29], STPA can be used as a method for establishing safety operations for UAVs with less than 150 kg (light UAVs) instead of traditional hazard analysis. [29] uses STPA having the control structure defined with a human pilot-centered approach and a certified aircraft. The authors in [30] and [31] made use of STPA during design as a method to develop a safer system in some particular applications as it will be done in this paper.

3. IFA²S Requirements Definition Using STPA Analytical Method

In this section, requirements for the development of IFA²S are defined using the STPA method. This method was chosen since it is considered ideal for ensuring that the new system will take into account several aspects potentially involved in an aerial accident, such as safety limits, aircraft components (health), meteorology, and environment (surrounding air traffic; cities and restricted areas on the route). In order to identify accidents and hazards in the operation of a UAV, the system is considered as composed by the aircraft, control station on the ground, a communication link between them both, and humans to control the aircraft and its payload. This work only considers collisions as an accident: A1. Collision with People on the ground; and A2. Collision with either another aircraft or with objects or property on the ground. The identified hazards that may cause these accidents are presented in Table 1.

Accidents and hazards related to UAV operation allow the definition of IFA²S requirements. A control structure for operation of the aircraft is presented in Figure 1, where control actions are represented by top-down vertical arrows and feedbacks are represented by vertical arrows in bottom-up direction. This structure allows the understanding of the flow of information and different aspects that may influence the flight. Mission commanders may provide the pilot with guidelines regarding the mission and pilot provides feedback regarding both mission and aircraft status. The pilot receives information using displays and

Table 1. Identified hazards.

H1	UAV infringes separation limits from another aircraft.
H2	UAV exceeds the flight envelope, such as speed and roll angle.
H3	UAV exceeds the operation envelope, such as vertical and horizontal limits.
H4	UAV fly over populated or sensible areas, such as military, nuclear plant, forest with fire.
H5	UAV is unable to continue flying due to internal health or weather.

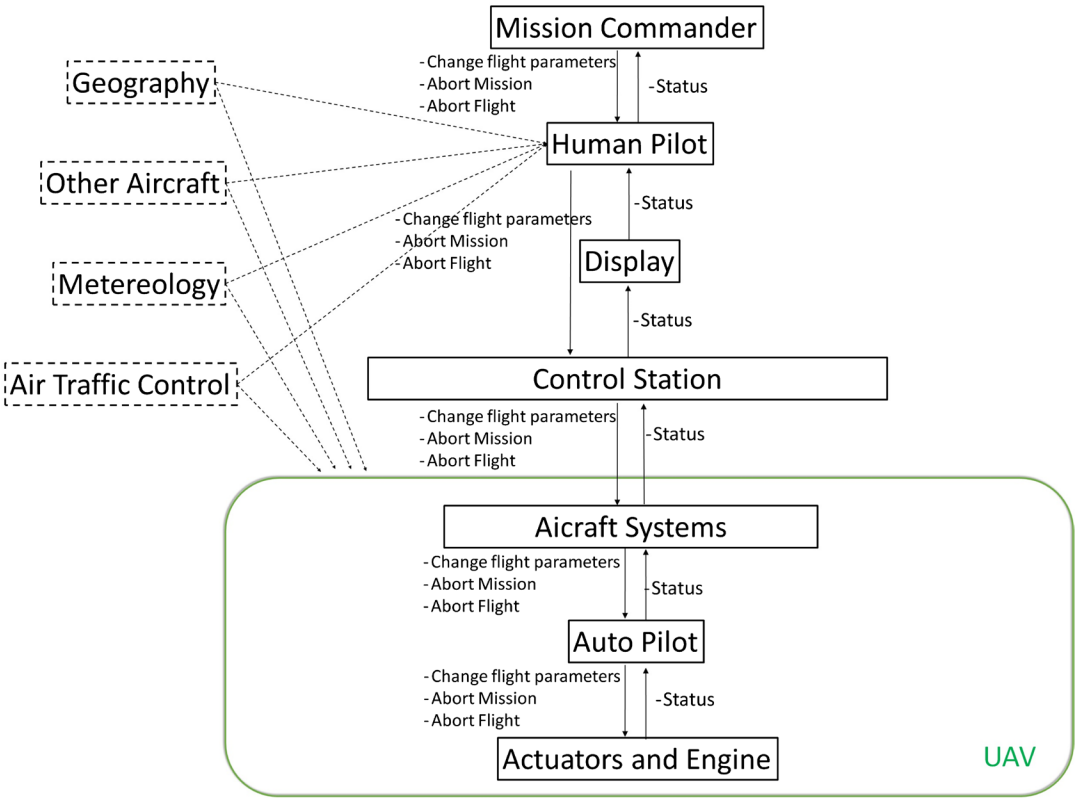

Figure 1. UAV control structure.

may act changing flight parameters or aborting the flight if an emergency occurs using control station interface. Orders are sent to aircraft by the station using some sort of radio link and telemetry is received to update aircraft health status. Inside the aircraft, the autopilot commands navigation in accordance with orders entering via radio system and eventual directives from other aircraft systems, such as direction change order from the S&A system. The autopilot controls the aircraft by changing actuators position and engine power. External influences may come from geography, aircraft in the proximity, meteorology, and traffic control. In accordance with the STPA method, hazards and the control structure create contexts and allow the identification of possible unsafe control actions.

The next step using STPA method is to identify scenarios and causal factors to understand how unsafe control action can arise. In this step, causal factors leading to hazards are identified by a scenario. In Figure 2, potential causal factors that can lead to hazards are highlighted in red and inadequate operation may come from hardware, software, human command, as well as a result of the interaction of any of these elements.

A fictitious scenario is presented as an example. Let's suppose the aircraft speed increases during a flight in order to accomplish mission schedule and, at a certain point, the UAV is close to its speed's upper limit, above which a structural damage may result. Looking at the display, the human pilot notices the

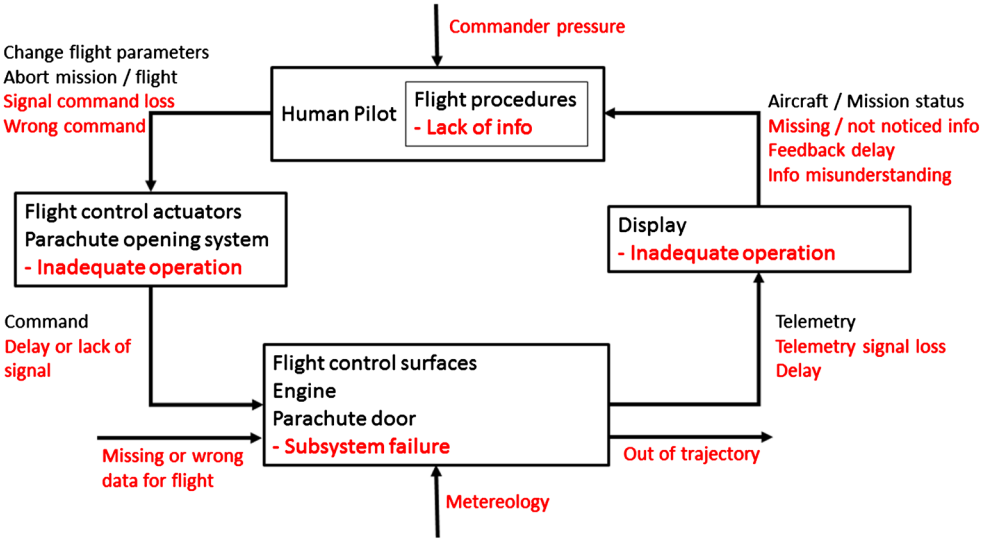

Figure 2. Control loop to identify the causal factors in hazards.

situation and commands an engine power decrease; nevertheless, the aircraft speed becomes even higher than top limit. Some causal factors could be: 1) Due to a bad design of the control station, the pilot ordered an increase in the speed instead of a decrease, as desired; 2) The command sent by the pilot was correct, but the order arrived too late in the engine actuator; 3) There was a loss in the radio link between the control station and the aircraft; and 4) There was a too long delay for the pilot to notice that the aircraft speed was close to its top limit.

Once identified scenarios and causal factors, requirements for IFA^{2}S could be established. These requirements are used to define the new system and ideally keep the aircraft away from hazards. For each scenario and causal factor mapped, a new feature is added to IFA^{2}S.

4. IFA^{2}S Modeling

IFA is the general framework for the design of the onboard system IFA^{2}S. The IFA^{2}S can be designed in many different ways, in accordance with requirements established with STPA and restrictions and objectives of a particular situation and aircraft. IFA dimensions provide data input to the decision algorithms: time, airworthiness, flight conditions, and information from the rest of the world. Time is an important dimension since an event has different meanings based on the moment it occurs. Airworthiness is defined as the safety in the operation of an aircraft and has some components: certification, manufacture, and maintenance. Flight conditions refer to the aircraft route: weather conditions, local air traffic, and overflown terrain. The information from the rest of the world can provide data that may affect the flight, such as a new political border, a volcano explosion, or a fire in the woods. Moreover, depending on specific situations, additional sensors may be integrated into the UAV to provide IFA^{2}S with relevant information to assure flight safety.

IFA²S system acts as a supervisor and communicates with the autopilot or directly to emergency actuators depending on the circumstances. In order to model flight safety improvement due to IFA²S, it is necessary to consider that the likelihood of hazards depends on the operational conditions as well as the concentration of humans and buildings on the ground. Useful models are described in [32] for establishing safety levels and consider collisions on the ground and in the air. These models evaluate material and human consequences in specific operational situations. For the purpose of this work, the model in [32] is modified in two different ways: 1) a range of values is considered for UAV reliability, instead of a single one; and 2) only one UAV is considered and it has a IFA²S system onboard.

It is usual to verify that small UAVs do not have reliability figures established for some or most of their components and parts and uncertainty may be classified as epistemic in this case. This work uses Dempster-Shafer theory, [33] [34], as proposed in [35] to deal with these epistemic uncertainties for evaluating aircraft failure rate λ. Instead of using precise figures, a reliable function is used to assess inaccurate data of components when evaluating flight conditions as in H5, **Table 1**. This approach provides a range of values for the UAV reliability. The expected ratio of collision F_C adapted from [32] is given in Equation (1).

$$F_c = \rho_o \phi_{col} V_{rel} \left(1 - \epsilon_{ifa,air}\right), \tag{1}$$

where: ρ_o: Aircraft total density (aircraft number/mission volume); ϕ_{col} : Total collision area; V_{rel} : Relative speed between the UAV and intruder; $\epsilon_{ifa,air}$: probability to avoid collision with intruder when IFA²S is onboard.

Whereas the IFA²S introduce mitigation mechanisms, the collision rate with people and buildings may change by different ways, depending on the probability of what is avoided: catastrophic failure due to an internal system ($\epsilon_{ifa,si}$), either a populated or a prohibited area and land in cruise flight conditions ($\epsilon_{ifa,geo}$), a catastrophic failure due to meteorology ($\epsilon_{ifa,met}$), or loss of control due to lack of communication link ($\epsilon_{ifa,fc}$). For simplicity, all factors are considered identical and equal to $\epsilon_{ifa,g}$. In this case, collision rates become:

$$F_{pf,p} = \lambda \sigma_p A_{LHp} \left(1 - \epsilon_{ifa,g}\right)^4, \tag{2}$$

$$F_{pf,b} = \lambda \sigma_b A_{LHb} \left(1 - \epsilon_{ifa,g}\right)^4, \tag{3}$$

where: σ_b, σ_P: Respectively, buildings and pedestrians density in the area (items/m²); λ: Failure rate for a single UAV (failures/hour), derived from its FTA; A_{LHP}, A_{LHb}: Respectively, lethal areas for pedestrians and buildings in emergency landings; $F_{pf,p}$: Collision rate due to collisions of the UAV with people; $F_{pf,b}$: Collision rate due to collisions of the UAV with human constructions.

Equations (4) and (5) present the rate of collisions with people and buildings due to a collision of the UAV with another aircraft in the air. The total rate of collision with persons and buildings can then be described as found in Equations (6) and (7). **Figure 3** presents the fatality rate change due to IFA²S. In this figure,

as $\epsilon_{ifa,g}$ varies from zero (no IFA²S onboard) to one, the probability of a fatal accident with a person on the ground drops noticeably. Curves refer to people concentration per area unit.

$$F_{ar,p} = F_c \sigma_p A_{LVp} \tag{4}$$

$$F_{ar,b} = F_c \sigma_b A_{LVb} \tag{5}$$

$$F_p = F_{ar,p} + F_{pf,p} \tag{6}$$

$$F_b = F_{ar,b} + F_{pf,b} \tag{7}$$

Following sections describe situations identified by STPA as potential causes for hazards and the methodology used to deal with them.

1) *Air Collision, H*1

Monitoring nearby aircraft requires a system to identify their position, speed, and bearing and an algorithm that allows an opportune diversion. For the purpose of this study, it was considered UAV has the means to identify nearby traffic.

Let $\mathbf{r}_{uav}(t)$ and $\mathbf{r}_{int}(t)$ respectively represent the position of the UAV and the intruder in the two dimensional space as functions of time *t*. A route conflict occurs if the distance between them becomes smaller than a value r_{lim} as in (8) and there is a too small vertical separation v_{lim}.

$$\left\| \mathbf{r}_{uav}(t) - \mathbf{r}_{int}(t) \right\| \le r_{lim} \tag{8}$$

For simplicity, r_{lim} was calculated for horizontal distance and an additional requirement was considered for vertical separation. Horizontal distance is calculated using Haversine formula, (9) and (10), where φ_{int} and λ_{int} are the latitude and longitude of the intruder, φ_{uav} and λ_{uav} are the latitude and longitude

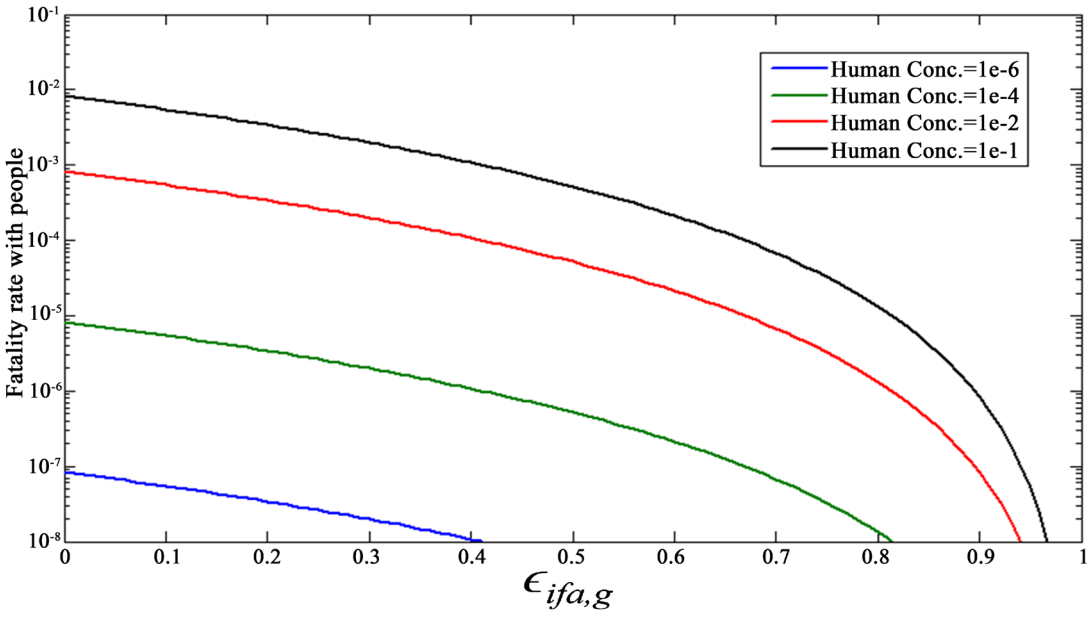

Figure 3. $\epsilon_{ifa,g}$ alters the fatality rate differently with changes in human concentration on the ground, $\lambda = 5 \times 10^{-3}$.

of the UAV, and R is earth's radius (6371 km).

$$r_{lim} = 2R\sin^{-1}(\Gamma) \tag{9}$$

$$\Gamma = \sqrt{\left(\frac{(\varphi_{int} - \varphi_{uav})}{2}\right) + \cos(\varphi_{int})\cos(\varphi_{uav})\left(\frac{(\lambda_{int} - \lambda_{uav})}{2}\right)} \tag{10}$$

In the case r_{lim} and v_{lim} are both smaller than established limits, UAV identifies whether the intruder is on either the left or on the right side in relation to its current heading and performs maneuver accordingly.

2) ***Bad Weather, H2, H3***

Meteorology is a major cause of air accidents and the lack of pilot onboard makes the situation even more complicated because of the need for additional monitoring. The strategy for the identification of turbulence presence was to establish limits for dimensions most impacted by its effects: vertical altitude and roll angle variations. Measurements were performed on the variable ξ (either vertical speed, $\dot{h}$, or roll angle, ϕ) and verified if 20 consecutive standard deviation measurements, N_T, are above a certain level, N_T^{Lim}. Number of samples was determined empirically during simulations, taking into account the capability of the system for identifying turbulences efficiently, avoiding both false alarms and aircraft dangerous attitudes as defined in (11) and (12).

$$\sigma = \sqrt{\left(\frac{1}{40}\right)\cdot\sum_{i=1}^{40}(\xi(i)-\mu)^2}\,;\ \mu = \left(\frac{1}{20}\right)\cdot\sum_{i=1}^{20}\xi(i) \tag{11}$$

$$N_T = \sqrt{\left(\frac{1}{20}\right)\cdot\sum_{i=1}^{20}(\sigma-\mu_\sigma)^2}\begin{cases}\geq N_T^{Lim} \Rightarrow \text{Alert}\\ < N_T^{Lim} \Rightarrow \text{Not Alert}\end{cases} \tag{12}$$

Turbulence was modeled with zero average white Gaussian noise over both amplitude $A_k(t)$ and phase $\theta_k(t)$ for each perturbation input in Equation (13). For the purpose of this work, an increase in σ corresponds to a turbulence augmentation applied to the *jth* individual control surface input u_j.

$$u_j = \sum_{k=1,\cdots,M} A_k(t)\sin\left(\left(\frac{2\pi kt}{T}\right) + \theta_k(t)\right), \tag{13}$$

where M represents the number of available frequencies and T is the excitation time.

3) ***Low Altitude Auto Recovery, H3***

Low altitude auto-recovery is performed as soon as the altitude of the aircraft is lower than a certain limit d_{lim}. This limit is established considering aircraft performance and terrain characteristics. Let h_{min} be a minimum acceptable distance from the ground, ψ_{prf} be the aircraft performance variable, and ψ_{ter} be the terrain altitude, all positive integers, then d_{lim} is given in (14).

$$d_{lim} = h_{min} + \psi_{prf} + \psi_{ter} \tag{14}$$

4) ***Avoiding Overflight of Forbidden Areas, H4***

Avoiding overflight of forbidden areas may obligate a new route to be set. The

motivation for avoiding flying over some areas arises from the recognition that UAV must not cross regions with either high population density or sensitive facilities, such as nuclear plants and military bases. This is a realistic scenario for UAV and we have a non-convex path planning problem with stay in and stay out areas as described in [36] using a mixed integer programming model.

For automatic rerouting, a greedy heuristic (GH) is applied as described in (16). The distance from forbidden areas was determined using Equations (15) and (16).

Let's set points A and B as the line limits of an area border, $\boldsymbol{v}_1$ the vector from A to aircraft, $\boldsymbol{v}_2$ the vector from B to aircraft, $\boldsymbol{w}$ the vector from A to B, and θ_1 and θ_2 the $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ angles in relation to $\boldsymbol{w}$, respectively. If $\theta_1 < 90^\circ$ and $\theta_2 > 90^\circ$, the distance d from aircraft to closest point P on the area border may be found.

$$\boldsymbol{v}_1 \cdot \boldsymbol{w} = |\boldsymbol{v}_1||\boldsymbol{w}|\cos\theta_1 \tag{15}$$

$$\boldsymbol{v}_2 \cdot \boldsymbol{w} = |\boldsymbol{v}_2||\boldsymbol{w}|\cos\theta_2 \tag{16}$$

$$d = \frac{|\boldsymbol{w} \times \boldsymbol{v}_1|}{|\boldsymbol{v}_1|} \tag{17}$$

When d was smaller than a limit, r_{lim}, the GH was used to determine a new route using (18), whose elements are described in Section 5. Figure 4 presents an example of a route determined using (18).

$$fitness = f_{L\phi b} + f_{L\phi n} + f_{DLand} + f_{Viol} + f_{\psi}, \tag{18}$$

5) *Emergency Landings, H4, H5*

Emergency landings also demand reroute, since it requires a safe place to land.

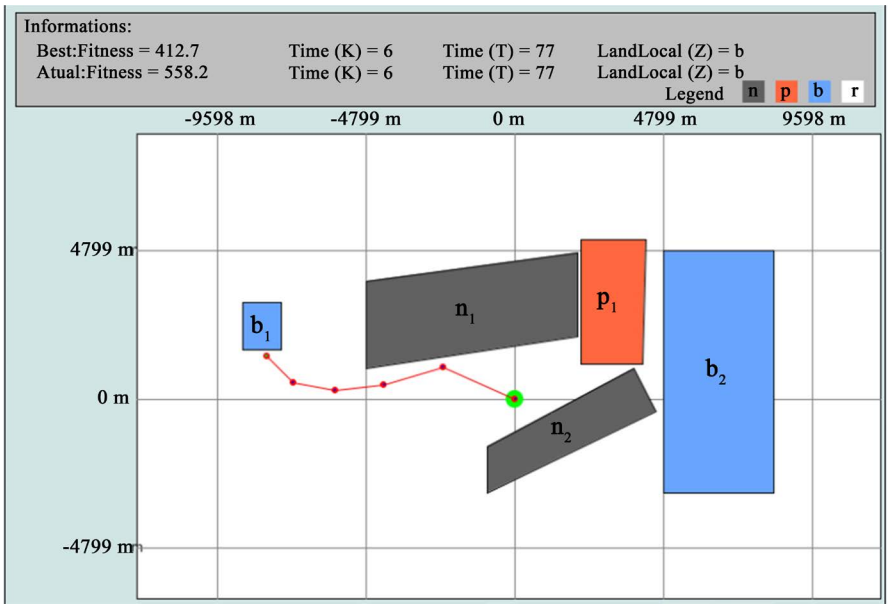

Figure 4. GH determines avoiding some regions on the ground and choosing bonus regions (b_n).

In this case, the genetic algorithm (GA) proposed in [25] [26] was used for emergency landings with fitness function given in Equation (19). This approach deals with uncertainty for aircraft position, so a variance of 10 m is set in its covariance.

$$fitness = f_{L\phi b} + f_{L\phi p} + f_{L\phi n} + f_{Curves} + f_{DLand} + f_{Viol} + f_{\psi} \tag{19}$$

where ϕ_n, ϕ_p, and ϕ_b are areas, respectively, not navigable (n), navigable with penalty (p) and navigable with bonus (b). The set $Z_{\phi j}$ has regions $j = \{n, p, b\}$; $C_{\phi j}$ is the cost of land in area ϕ_j; Δ is the likelihood of the UAV violate area ϕ_j; u_t is the control set; ε_t is the angular variation of the UAV at time t on the x axis; x_t is the position at instant t; K is navigation time that shall be smaller than a limit T_{lim}.

Equation (20) defines reward in case of landing in bonus set regions. Equation (21) defines punishment in case of landing in penalizing regions. Equation (22) penalizes landing or flight of the aircraft on non-navigable areas. Equation (23) prioritizes routes that avoid making unnecessary curves. Equation (24) gives more chances to routes with smaller distances from bonus set regions. Equation (25) prevents routes where the UAV cannot fly over, even if it allows reaching a bonus set region. If there is a problem in the battery, (26) is added to the fitness function to reduce the flight duration. Details about all these functions are reported in [25] [26].

$$f_{L\phi b} = -C_{\phi b} \sum_{i=1}^{\phi b} P\left(x_k \in Z_{\phi b}^i\right) \tag{20}$$

$$f_{L\phi p} = C_{\phi p} \sum_{i=1}^{\phi p} P\left(x_k \in Z_{\phi p}^i\right) \tag{21}$$

$$f_{L\phi n} = C_{\phi n} \max\left(0, 1 - \Delta - P\left(\wedge_{t=0}^{K} \wedge_{i=1}^{|\phi n|} x_t \notin Z_{\phi n}^i\right)\right) \tag{22}$$

$$f_{Curves} = \left(\frac{1}{|\varepsilon_{max}|}\right) \cdot \sum_{t=0}^{K} \|u_t\| \cdot |\varepsilon_t| \tag{23}$$

$$f_{DLand} = lowerDist\left(\underline{x}_K, Z_{\phi b}^i\right) \tag{24}$$

$$f_{Viol} = \begin{cases} C_{\phi b}, & \text{if } v_K - v_{\min} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

$$f_{\psi} = \begin{cases} C_{\phi b} 2^{\frac{K-T}{10}}, & \text{if } v_K - v_{\min} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{26}$$

6) *Failure in Aircraft Systems, H5*

In this study, only three systems were elected to serve as case study for the development of concepts and the state machine: Low battery voltage; High battery temperature; and High current in the avionics system. The actions triggered can be returned to base, emergency landing, or parachute opening (**Figure 5**). IFA²S decisions depend on the measured values and the limits established by the operator using his/her interface (**Figure 6**).

5. Simulation Results Using LabView and XPlane

Given the complexity of UAV operations and the difficulty to establish satisfactory formats and algorithms for embedded systems, the simulation environment allows testing ideas while complying with requirements. IFA²S was codified using Labview, Figure 6, and flights using XPlane simulator. GH method was implemented in C, using SCADE Suite® and GA in java; both were external codes called by Labview software.

A state machine was created to allow IFA²S solutions assessment, Figure 5. During the flight, IFA²S system remains in state "*Idle*" until an event leads to a state change. These events may be generated in XPlane due to flight conditions, e.g., bad weather, or within the Labview such as failure in the aircraft battery for example, since it is not easy to engender this kind of occurrence inside XPlane. The interface created, Figure 6, enables the user to force events to occur in a controlled way and to verify the results of the algorithms under analysis. As an example of how different events may have similar effects and be misunderstood by the system, in a test of reactions to low altitude (when the aircraft shall increase its altitude as fast as possible), the system interpreted these fast variations

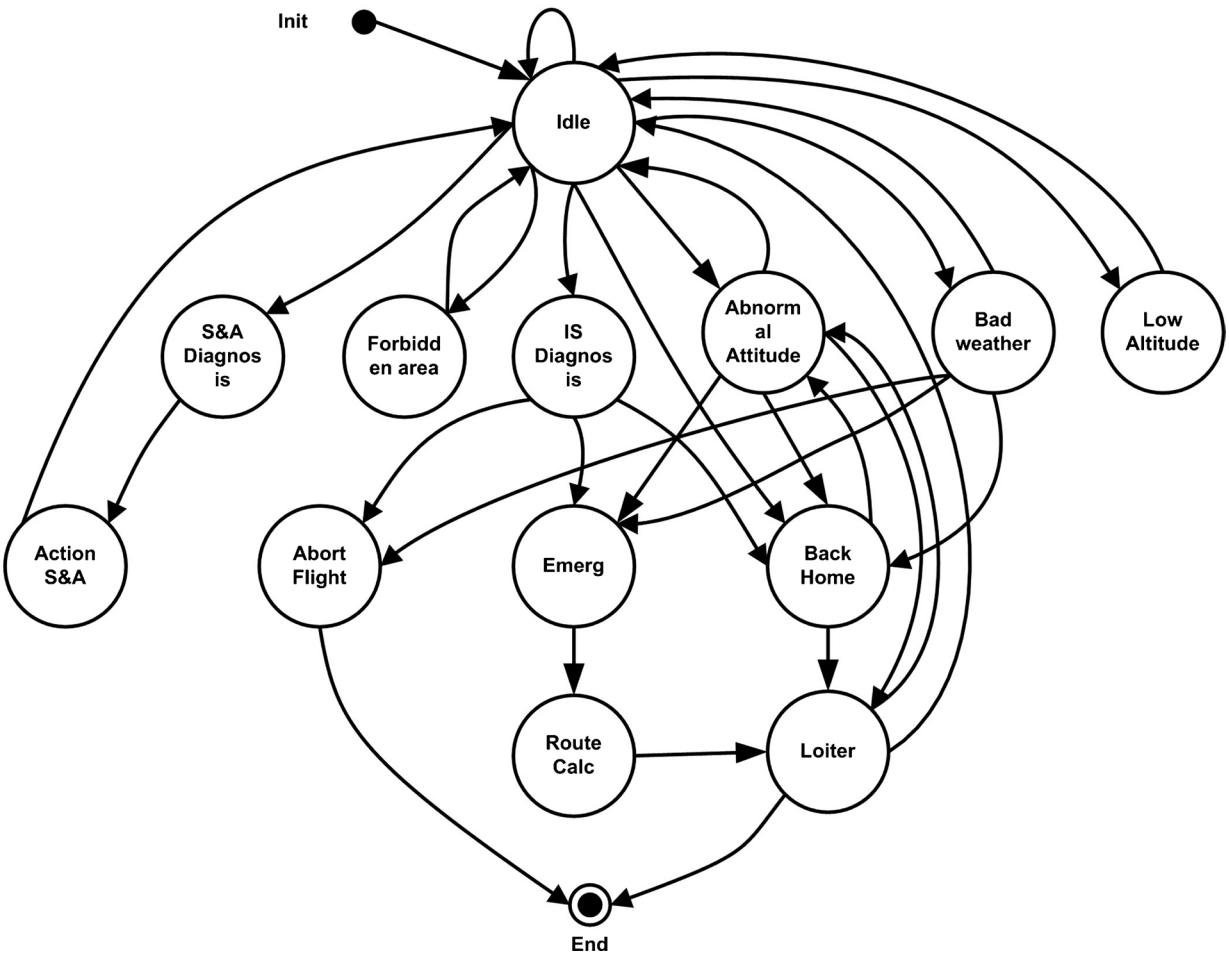

Figure 5. State machine implemented at Labview for IFA²S assessment.

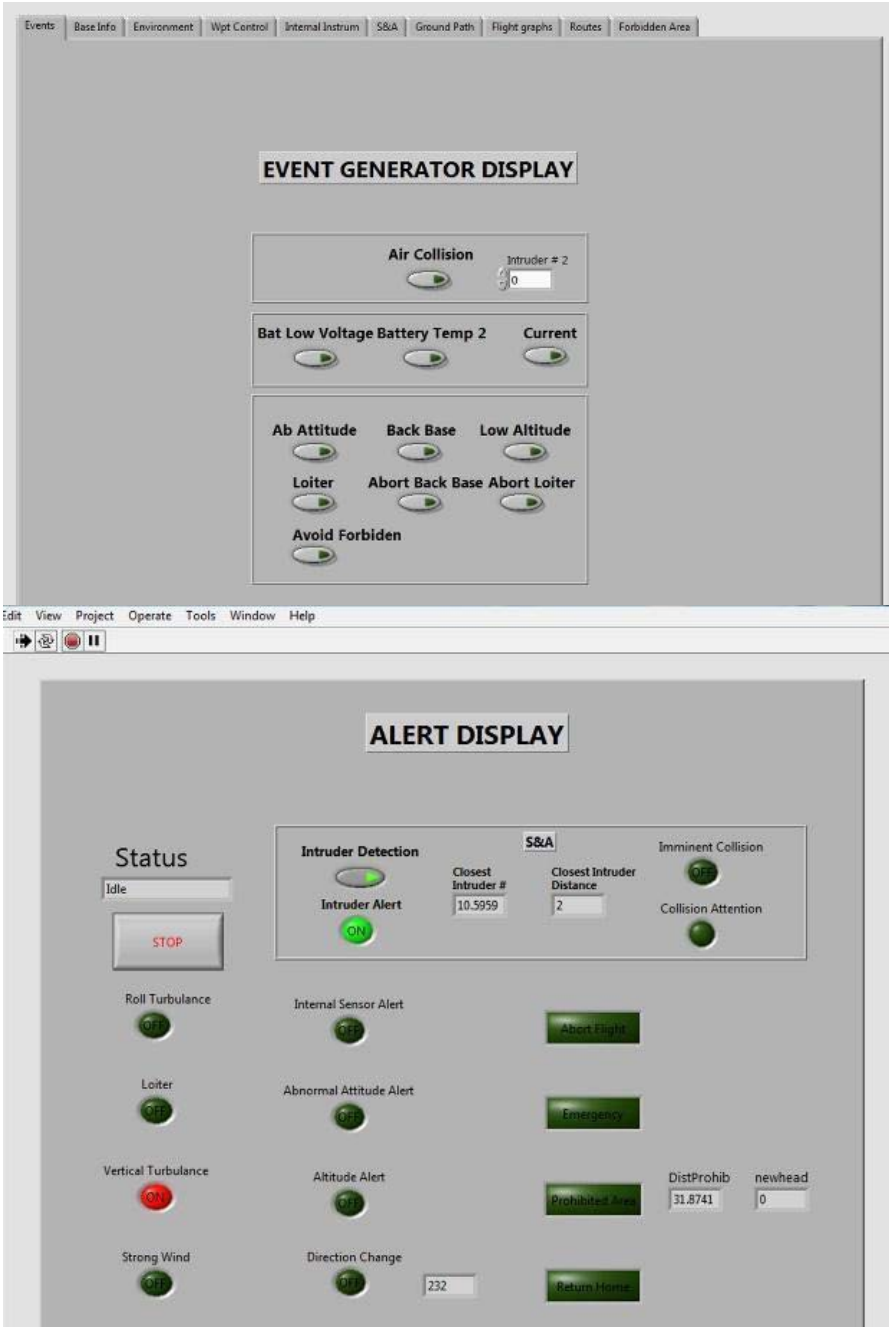

Figure 6. Labview interface used to control and monitor IFA²S behavior.

in the vertical speed as turbulence. Misinterpretations may lead to wrong decisions.

1) *Failure in Aircraft Systems*

XPlane does not allow the creation of failure in aircraft systems, thus they are engendered within the Labview event generator, Figure 6. The tests in simulated flight, Figure 7, are consistent with the logic adopted by the state machine and a total of 20 experiments were accomplished varying limits to verify states change and behavior. The state machine acted as expected ("*Idle*", "*Return Home*", or

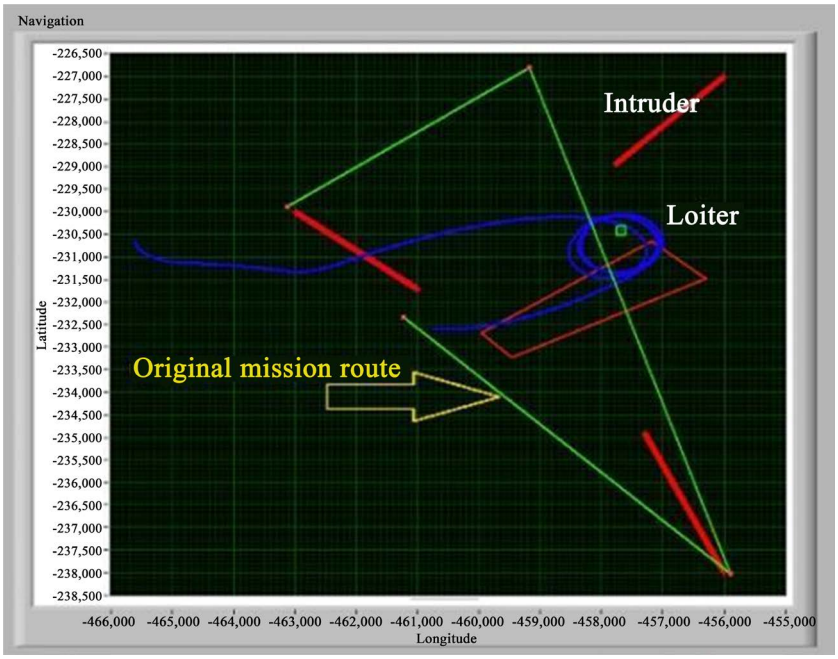

Figure 7. A failure on aircraft systems may cause return to Base (square green point), followed by a descending loiter pattern (blue line). Operators may resume action and send UAV to resume the original mission route (green lines).

"*Loiter*") and all were successfully handled by IFA²S. The operator could cancel the return to base action at any moment and, in this case, the original mission was resumed.

The command "*Return Home*" causes the aircraft to navigate to the coordinates of the base. Once the aircraft reaches a distance of 2 km of the base coordinates, state is changed to "*Loiter*" and a descending circular flight is started until 1000 ft (305 m) altitude from the ground. In this situation, the aircraft stops descending and maintains altitude until there is the command "*Abort Loiter*" by the operator, available in the "*Event Generator*" window. The operator can always cancel automatic actions during operation.

2) *Air Collision*

Monitoring nearby aircraft requires a system to identify their position, speed, and bearing and action an algorithm that allows an opportune diversion. For testing this functionality inside the state machine, simulated aircraft were created in the operation area, called intruders, and a collision course was set to verify if the intended course change was capable of avoiding mishap in time.

Once the test was selected, the UAV headed directly to the nearest intruder aircraft. As the intruder aircraft cross both vertical and horizontal limits, an alert is displayed to the operator and the aircraft changes its route autonomously. The limits set forth in this simulation were 5 km to change route (red alert), 10 km to yellow alert, and 152 m for vertical clearance. Once the intruder was outside the emergency limits, the UAV resumed its original mission.

In general, the algorithm used to avoid collisions in flight worked as expected,

being able to avoid an excessive proximity between aircraft. Out of 10 experiments, starting maneuver when closer than 5 km, the minimum distance found was 2.2 ± 0.4 km, Table 2.

3) ***Bad Weather***

The identification of atmospheric turbulence autonomously requires a better understanding of its effects on the aircraft. It mainly causes random variations in the roll angle and vertical axis. Variations in the vertical axis are changes in the climb rate ($\dot{h}$) and not pitch angle. XPlane has its own interfaces to control turbulence conditions in the longitudinal and transverse axes independently. Only roll angle was considered in (13) and the measured XPlane's turbulence spectrum presented only one harmonic component (M = 1).

Measurements were made using one-hour flights at each turbulence level, starting with no atmospheric instability (named W0) and increasing it gradually (W1, W2, and W3). In other words, the state W0 means no turbulence was allowed during the flight and it was at its maximum level at W3 with two equally spaced intermediate stages (W1 and W2). Table 3 shows the time in minutes needed to trigger an action from IFA^{2}S in different situations.

Figure 8 shows the appearance of a minimum threshold as turbulence is increased for climb rate measurement, a similar figure was found for vertical speed. Since real world conditions do not separate both axes and aim to reduce

Table 2. Results for avoiding collisions in flight.

Intruder relative position	UAV-Intruder minimum distance (km)	Maneuver time (s)
Right	2.2	32
Right	2.5	36
Right	1.8	27
Right	2.1	32
Left	1.7	26
Left	2.8	40
Left	2.1	32
Average	**2.2**	**32**
Standard Deviation	**0.4**	**5**

Table 3. Time (min) to adverse atmospheric condition identification using minimum limits variations of the standard deviation (σ) of both vertical speed and roll angle.

Roll σ (degrees)	Vert speed σ (m/s)								
	500			1000			1500		
	W1	W2	W3	W1	W2	W3	W1	W2	W3
0.015	51	7	3	51	27	5	-	31	44
0.020	-	7	4	-	-	19	-	-	44
0.025	-	14	4	-	-	19	-	-	44

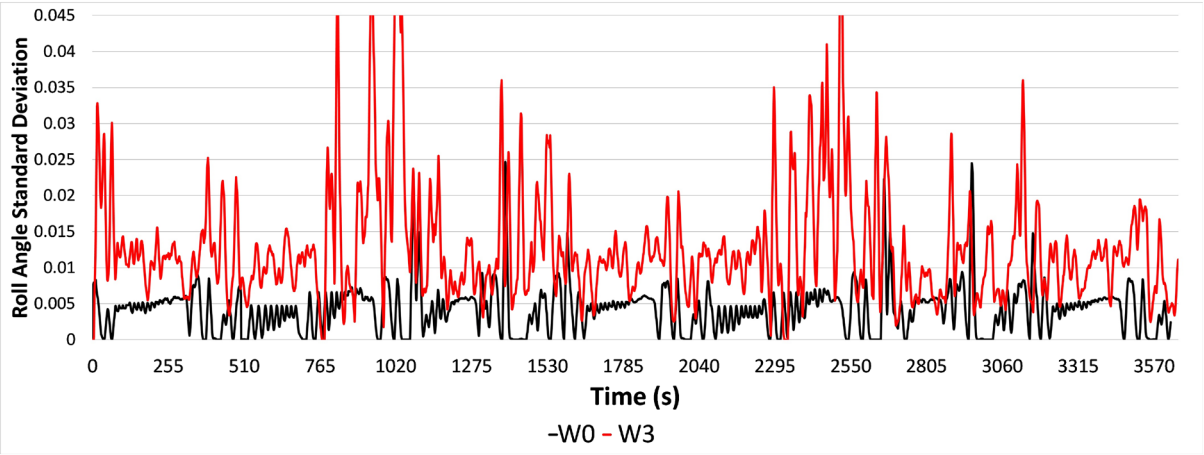

Figure 8. Standard deviation of the roll angle in two one-hour flights in turbulent conditions levels W0 (black) and W3 (red).

false alarm rate, a new instrument was incorporated at the Labview operator interface, named "turbulence meter", to count the number of times certain limits were surpassed in both axes.

Actions depend on the number of occurrences at the "turbulence meter". The first limit defines that the weather conditions are bad and makes the aircraft go back home. The second limit provokes an emergency landing. A third limit defines the situation as "too dangerous" and the aircraft opens the parachute.

Another result for these flight tests was to realize that the aircraft was much more sensitive to roll angle than to vertical variation. The control system limited the roll angle in 40 degrees, but weather instabilities triggered higher values, depending on its intensity. If the aircraft roll angle was greater than the limit of 60 degrees, the state "*Abnormal Attitude*" was triggered and lasted on average for 1 - 2 seconds for stabilizing position (zero degrees for roll, yaw, and pitch angles). A total of 20 controlled tests were performed successfully. All attempts to use the level W4 resulted in the state "Abort Flight" with consequent parachute opening. Whereas flight tests did not identify W0 and W1 as threats to safety, the system was considered as sufficient turbulent flight to return to base those classified as W2 and W3.

4) ***Low Altitude Auto Recovery***

The altitude recovery mode abandons the ongoing mission and puts the aircraft in the attitude recovery mode (1 - 2 seconds for stabilizing position: zero degrees for roll, yaw, and pitch angles) and the aircraft starts an upward flight with maximum power applied to the engine. A constant verification of altitude in relation to the ground is performed during the flight in the states "*Idle*", "*Back Home*", and "*Loiter*" and, if the altitude is less than 1500 ft (457 m), (14), "*Low Altitude*" state was enabled in order to avoid a catastrophic collision with the ground. Out of 18 experiments, IFA^{2}S was capable of starting to recover altitude after 67 ± 45 ft, as may be noticed in **Figure 9**. In future work a more complex model to avoid obstacles may be developed, such as used by [37] [38].

5) *Avoid Overflight of Forbidden Area*

The solution to avoid overflight of a prohibited area was described in Section 4—4). by incorporating in IFA²S the GH implemented in C code as re-routing planner, Figure 10. We are proposing a path planning for non-convex scenarios, which was dealt with by [25] [26] [36]. Ten tests were pursued and the aircraft

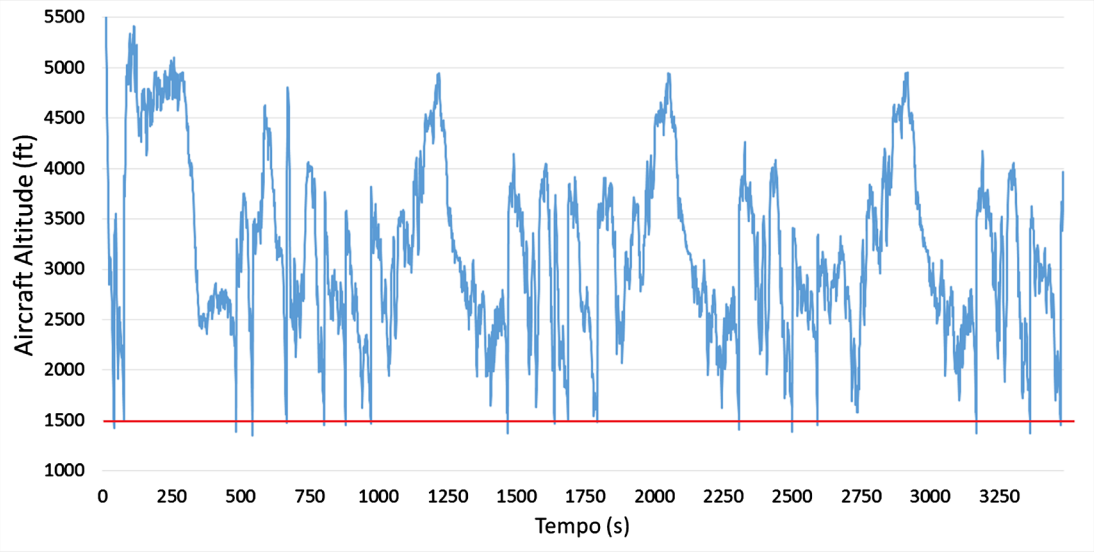

Figure 9. The state "*Alt*" was triggered whenever the aircraft altitude was smaller than 1500 ft (457 m).

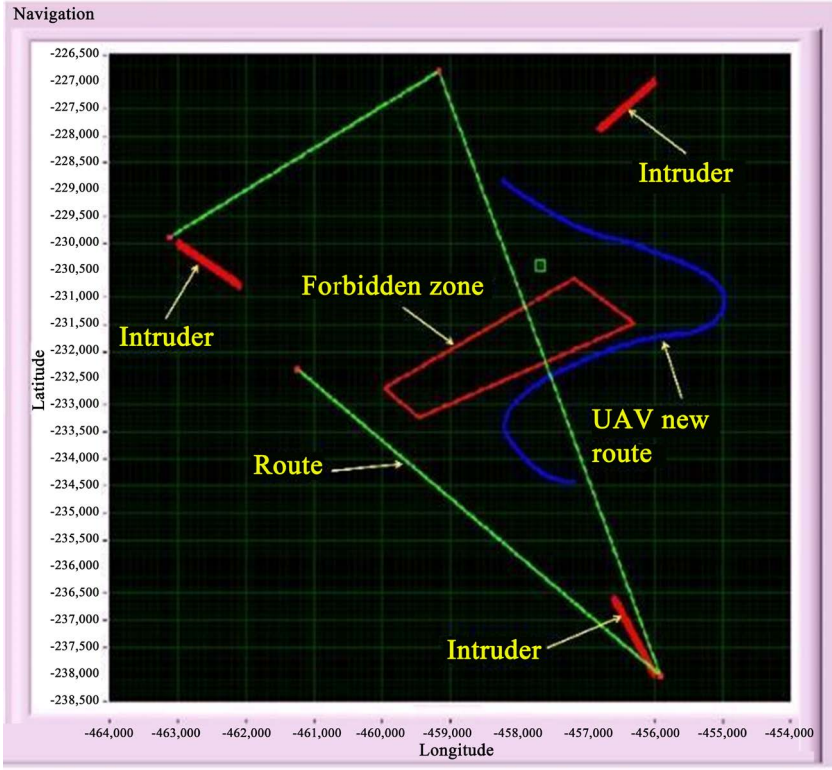

Figure 10. Automatic route deviation to avoid flying over a forbidden zone during simulation using XPlane flight simulator and Labview.

avoided the overflight in 8 attempts. In both unsuccessful simulations, although the aircraft avoided flying over the forbidden area, it penetrated ± 2 km into its limits as part of the maneuver. Despite the small depth of penetration, the aircraft traveled a distance around 7 km before leaving the area.

The test to avoid overflight of the forbidden zone started when the aircraft was closer than 10 km to the area's borders. In this case, a new route was generated. Once the aircraft was farther than 15 km, it resumed navigation and the alert lights were turned off by IFA^{2}S.

6) ***Emergency Landing***

In the event of a failure, opening of the parachute is an option to avoid a catastrophic crash on the ground, but it may result in damage or even total loss of the aircraft. One option to parachute opening is the aircraft to seek a place where the chances of an automatic landing can be accomplished with minimal chances of damage to people, property, or itself. This feature has been incorporated into the IFA^{2}S as an alternative in specific situations using the GA to plan a route for emergency landing as described on Section 4—5).

Two thousand simulations were carried out in ten different situations and the algorithm was capable of creating a new path in 3 s by reaching the final waypoint in less than 3 min. A test was considered successful when the final waypoint was within an appropriate zone for emergency landing. In Table 4, it may be noticed that the aircraft bearing influenced the results obtained due to position in relation to the zones surrounding it. The best situation is the one in which there was only one attractive field and no need to curve to avoid a forbidden zone. In the worst case scenario, test number 10, the aircraft was too close and heading directly to a zone where it was not supposed to overfly. Figure 11 shows an example of a solution provided by the GA.

7) ***Abnormal Attitude***

This state was used to correct excessive roll and pitch angles. The maximum

Table 4. GA simulation results varying aircraft heading.

Test #	Bearing (degrees)	Successes	σ x axis (m)	σ y axis (m)
1	270	100%	5	5
2	227	100%	6	8
3	270	59%	103	59
4	270	71%	95	125
5	73	79%	64	14
6	358	80%	65	18
7	335	84%	339	32
8	176	95%	12	14
9	101	87%	70	49
10	118	14%	-	-

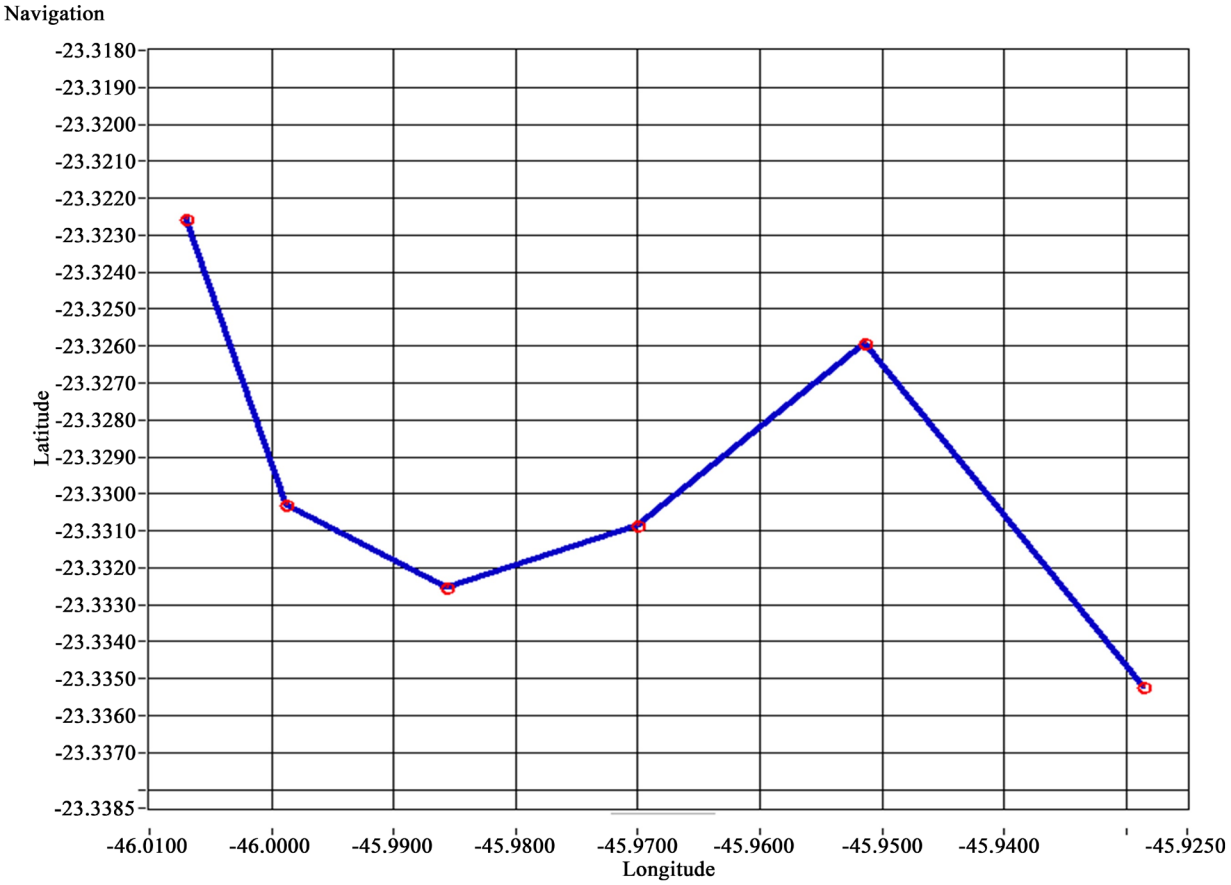

Figure 11. Example of emergency landing route calculated by the GA used by Labview to control flight inside XPlane, initial aircraft bearing equals 350 degrees.

angles defined at the control system were ±40 degrees for roll and ±10 degrees for pitch, however the presence of disturbing elements may cause values that exceed this limit and eventually trigger a loss of control state.

Once IFA²S, in states "*Idle*", "*Back Home*", or "*Loiter*", identifies a roll angle greater than 60 degrees, the state machine enters in the state "*AbA*" for abnormal attitude correction. Once the excessive angle is within limits, the original state is restored. If the roll angle is greater than 85 degrees or the pitch angle is greater than 50 degrees, the flight is aborted and the parachute opens.

Atmospheric turbulence was the most common origin for exceeding roll angle, which simulations and flights proved to be more sensible to turbulences than the pitch angle. In all tests, IFA²S has performed accordingly to the state machine and avoided loss of control or opened parachute.

8) ***Abort Flight***

Prompt flight termination is done by parachute landing, after a controlled motor stop. This action is intended to put the plane down in a close proximity with the previously intended flight path, avoiding excessive low altitude flight and minimizing the energy on the ground impact. This procedure reduces risks related to personnel injury, ground installations damage, aircraft damage, and

mission sensor loss. This functionality was engendered in the simulations by using a window at the operator's interface.

Figure 12 shows an example of a flight termination due to meteorological conditions. During flight tests, vertical climb rate exceeded the established limit of 10 m/s and the parachute was opened. In this case, the aircraft did not have an IFA²S and the operator did not notice the situation. The red circle indicates the moment when the parachute opened.

6. Flight Experiments

1) *System Description*

The implementation of a complete IFA²S in small UAVs depends on the available payload, electrical power, empty space, and cost restrictions. This is the case of Tiriba®, Figure 13, a small UAV jointly developed by the company AGX

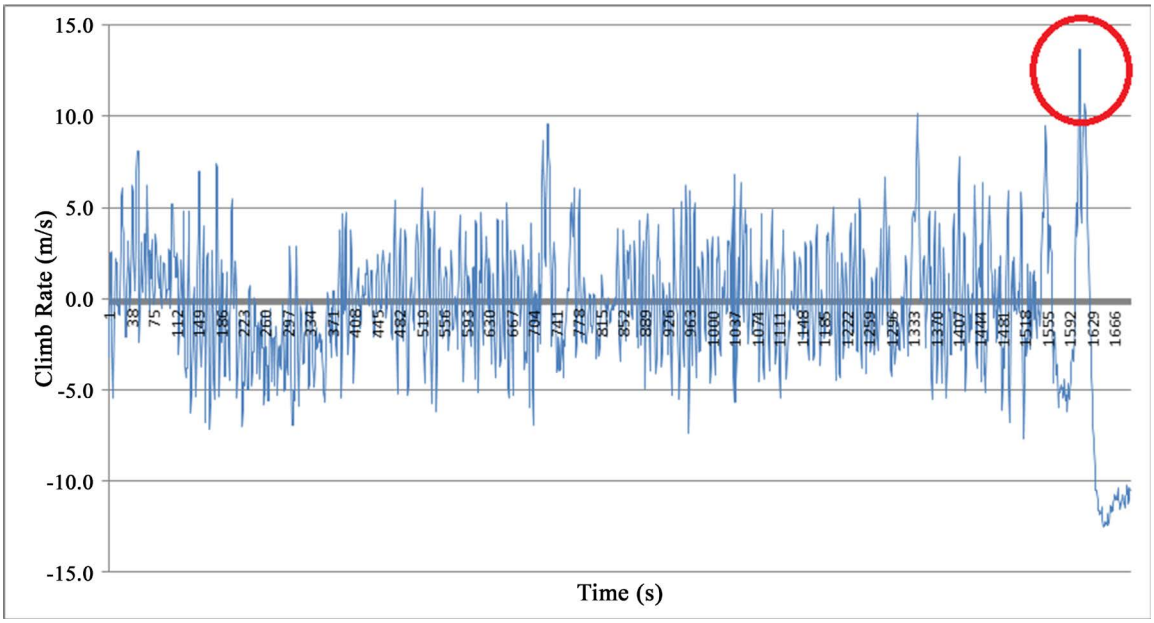

Figure 12. Climb rate variation during flight tests using an aircraft without IFA²S onboard. The operator did not notice the weather turbulence and parachute opened when the limit of 10 m/s was exceeded (red circle).

Figure 13. The UAV Tiriba® was used in flight tests.

Technology and University of Sao Paulo (USP). Tiriba is a hand-launched plane with wingspan of 2.2 m, electric engine, cruise speed of 110 km/h, maximum take-off weight of 4.5 kg, and endurance of 30 min.

Implementing IFA features in a small plane such as Tiriba is not an easy task since allowances are tight in space, electric power, weight, and cost. It is a first attempt and due to these limitations, IFA^{2}S was only partially implemented in Tiriba to test possible solutions aiming to comply with some few safety requirements. Simulations using the Labview software as well as a gradual implementation in small planes allow a progressive maturation aiming the development of a more complete system. Moreover, in order to avoid an additional controller on-board, IFA^{2}S was implemented as part of the autopilot software, running in the same hardware. This approach has some advantages: savings in power consumption, space, weight, and cost; easier system integration; and shared use of the available sensors. Some disadvantages are: no replicated sources for data validation and hardware have a single point of failure for both autopilot and IFA^{2}S. Table 5 correlates the IFA^{2}S implementation for Tiriba with IFA dimensions

Table 5. IFA Dimensions × Implementation × Requirements.

IFA Dimension & Actions	Tiriba Implementation	
	Developed	*Hazard*
Flight Conditions	Inferred by sensing abnormal flight atitudes:	
	- Max wing banking	H2
	- Max & Min climb rate	H2
	Or other abnormal flight conditions:	
	- Max & Min speed	H2
	- Max & Min altitude	H3
	- Max crosstrack error (inability to follow the programmed path)	H5
Airworthness	- Sensor failure	
	- General hardware fault (electronics)	
	- Nonsense data from sensors	
	- Dead engine	
	- Faulty communication with the ground station	H5
	- Low Battery	
	- Processor failure	
	- Software watchdog timer expiration	
	- Software fault preventing heartbeat generation	
Actions	- Send a warning to the ground station in dangerous situations	H2, H3, H5
	- Abort mission and return home when there is still flight conditions	H5
	- Terminate the flight promptly (emergency parachute landing)	H5

and hazards identified by STPA, Table 1. All sensed data come from the autopilot sensors.

Tiriba's fault tree analysis shows it has no redundant systems of any nature. Particularly, faults in actuators (such as servomechanisms) are indirectly detected through the sensing of abnormal flight attitudes. These abnormal attitudes can, otherwise, be related to severe weather conditions, such as strong winds. In such cases, it is difficult to evaluate if the flight must be terminated or not. In Tiribas' implementation, flight is always promptly terminated, favoring safety over mission accomplishment. Better sensing and better decision algorithms can avoid unnecessary flight termination.

In order to avoid the single point of failure represented by the autopilot hardware, there is an analog, high-reliability electronic board that controls the engine and the parachute. This board acts as a watchdog timer, receiving a heartbeat signal from the IFA^{2}S software. If for some reason the board stops receiving the heartbeat signal, it stops the engine and deploys the parachute. The same action is taken in the case of a complete power failure, since it is triggered by a spring/solenoid mechanism that must remain powered to keep normal flight operation. Figure 14 depicts a state machine of the watchdog board operation.

2) ***Results***

Table 6 presents some practical results from the first 100 flights using the preliminary IFA^{2}S version onboard Tiriba. Being a preliminary attempt, instead of using a separated board, IFA^{2}S algorithms were stored inside the autopilot processor. This situation is far from being ideal since a failure in this processor shall cause a loss not only in the autopilot but also in its supervisor (IFA^{2}S). Albeit this configuration shall be changed in the next trials, it allowed aircraft recovery as well as both the evaluation of this awareness improvement in flight safety and clues to make better decision mechanisms.

Responses to adverse conditions presented in Table 6 show interesting aspects of this setup. In a total of 100 flights, it may be seen two basic different cases: aircraft without and with IFA^{2}S. Additionally, when this supervisory system was

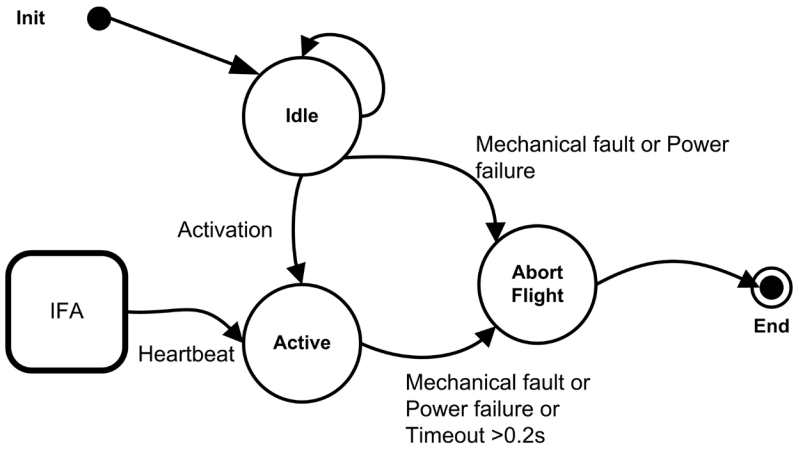

Figure 14. State machine of the watchdog board operation.

Table 6. Summary of occurrences from the Tiriba's first 100 flights.

Before IFA²S		
Flights performed successfully		20
Flights with loss of control: Lack of memory for dynamic allocation		4
Post IFA²S (occurrences with parachute opening)		
Flights performed successfully		56
IFA²S Action	Opening Parachute due to loss of control: Lack of memory for dynamic allocation	2
	Excessive roll angle (Adverse weather conditions)	3
	Climb rate too low	
	Power outage	2
	Descending air stream	1
	Descent rate too high	
	Assisted pilot mode doing too abrupt maneuvers	4
	Accidental parachute deployment	2
	Improper switching from automatic to assisted mode	
	Glitch in radio control used	4
	Aircraft structural failure	2
Post IFA²S (occurrences without parachute opening)		
IFA²S Action: Back home		6

onboard, two situations were examined: when the parachute was released and when it was not. Out of 100, 76 flights presented no problem, 20 before the incorporation of IFA²S, and 56 after. Before IFA²S, 4 losses were due to a bad memory management (lack of memory for dynamic allocation). After IFA²S, there was no damage to equipment and it was capable of identifying threatening situations and acting accordingly. What is important in the second case is that not only the aircraft was preserved but also the risk of harming someone on the ground or damaging a building was avoided.

The daily experience using Tiriba has revealed some important conclusions. First of all, the IFA²S has avoided plane loss in many dangerous situations ranging from battery faults, servomechanisms faults, structural faults, sensor faults, strong winds, and software bugs. On the other hand, it has terminated flight in conditions where the plane could recover itself and return to normal flight. These conditions include strong wind gusts as well as abnormal climb rates due to thermal ascending or descending airflow. Nevertheless, it is not easy to discriminate if this abnormal behavior has its origins in the atmosphere or on a broken servomechanism. Improvements in decision algorithms and sensors will reduce these cases, but tradeoffs between flight safety and the mission accomplishment shall remain.

7. Conclusion

The recognition that flight safety is the main barrier to the acceptance of UAVs

into airspace has driven the efforts for the development of means to increase its autonomy in risky situations. The development of the IFA2S system proved to be capable of avoiding accidents and hazards, via the increase of aircraft awareness and proper algorithms, based on both simulations and flight tests results. The safety requirements were properly defined by using STPA and proved to be satisfactory for the development of IFA^{2}S. STPA and the concepts regarding IFA model become possible to assure that IFA2S is able to monitor internal and external information, processes them, and acts in accordance with some technical and operational parameters. The simulations used a state machine created inside Labview to act as IFA^{2}S pursuing simulated flights inside XPlane flight simulator. The results obtained from the flight tests, using a preliminary IFA^{2}S onboard the UAV Tiriba, proved successful to improve capabilities to avoid some hazardous situations. Overall, the results met the objective of avoiding the critical situations identified and decreased the risks identified as safety requirements by STPA. The current main limitations of IFA^{2}S are the need for more flights in real world scenarios to validate the system performance. Once IFA^{2}S is employed, e.g. to accomplish missions in precision agriculture or surveillance scenarios, it will be possible to add improvement in the current state machine developed. Thus, as future work, it will involve improvements in the state machine for flight tests and more complex decision algorithms.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Radoglou-Grammatikis, P., Sarigiannidis, P., Lagkas, T. and Moscholios, I. (2020) A Compilation of UAV Applications for Precision Agriculture. *Computer Networks*, **172**, Article ID: 107148. https://doi.org/10.1016/j.comnet.2020.107148

[2] Albeaino, G., Gheisari, M. and Franz, B.W. (2019) A Systematic Review of Unmanned Aerial Vehicle Application Areas and Technologies in the AEC Domain. *ITcon*, **24**, 381-405.

[3] Nowak, M.M., Dziób, K. and Bogawski, P. (2019) Unmanned Aerial Vehicles (UAVs) in Environmental Biology: A Review. *European Journal of Ecology*, **4**, 56-74.

[4] Shakhatreh, H., Sawalmeh, A.H., Al-Fuqaha, A., Dou, Z., Almaita, E., Khalil, I., Othman, N.S., Khreishah, A. and Guizani, M. (2019) Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges. *IEEE Access*, **7**, 48572-48634. https://doi.org/10.1109/ACCESS.2019.2909530

[5] Claudia, S., Bennett, R., Nex, F., Gerke, M. and Zevenbergen, J. (2017) View of the Current State of UAV Regulations. *Remote Sensing*, **9**, 459. https://doi.org/10.3390/rs9050459

[6] Elsayed, M. and Moataz, M. (2020) The Impact of Airspace Regulations on Unmanned Aerial Vehicles in Last-Mile Operation. *Transportation Research Part D: Transport and Environment*, **87**, Article ID: 102480.

https://doi.org/10.1016/j.trd.2020.102480

[7] Pestana, M.E. (2011) Flying Unmanned Aircraft: A Pilot's Perspective. *AIAA Infotech at Aerospace Conference and Exhibit*, St. Louis, 29-31 March 2011, 1409. https://doi.org/10.2514/6.2011-1490

[8] Mirchandani, C. (2020) Cost-Effective Control of Unmanned Aircraft Systems. *AIAA Scitech* 2020 *Forum*, Orlando, 6-10 January 2020, 219. https://doi.org/10.2514/6.2020-0219

[9] Heydari, B. and Dalili, K. (2015) Emergence of Modularity in System of Systems: Complex Networks in Heterogeneous Environments. *IEEE Systems Journal*, **9**, 223-231. https://doi.org/10.1109/JSYST.2013.2281694

[10] Machuca, J.P., Miller, M.E. and Colombi, J.M. (2012) A Cognitive Task Analysis-based Evaluation of Remotely Piloted Aircraft Situation Awareness Transfer Mechanisms. *2012 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support*, New Orleans, 6-8 March 2012, 179-182. https://doi.org/10.1109/CogSIMA.2012.6188376

[11] Endsley, M.R. (1988) Design and Evaluation for Situation Awareness Enhancement. *Proceedings of the Human Factors Society* 32*nd Annual Meeting*, Anaheim, 24-28 October 1988, 97-101. https://doi.org/10.1177%2F154193128803200221

[12] Sayers, J.M., Feighery, B.E. and Span, M.T. (2020) A STPA-Sec Case Study: Eliciting Early Security Requirements for a Small Unmanned Aerial System. 2020 *IEEE Systems Security Symposium*, Crystal City, 1 July-1 August 2020, 1-8. https://doi.org/10.1109/SSS47320.2020.9197728

[13] Endsley, M.R. (1995) Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors*, **37**, 32-64. https://doi.org/10.1518%2F001872095779049543

[14] Endsley, M.R. (1996) Automation and Situation Awareness. Automation and Human Performance: Theory and Applications. Lawrence Erlbaum, Mahwah, 163-181.

[15] Drury, J.L., Riek, L. and Rackliffe, N. (2006) A Decomposition of UAV-Related Situation Awareness. *Proceedings of the* 1*st ACM SIGCHI/SIGART Conference on Human-Robot Interaction*, Salt Lake City, March 2006, 88-94. https://doi.org/10.1145/1121241.1121258

[16] Naderpour, M., Lu, J. and Kerre, E.A. (2012) Conceptual Model for Risk-Based Situation Awareness. In: Wang, Y. and Li, T. Eds., *Foundations of Intelligent Systems*, Vol. 122, Springer-Verlag, Berlin, 297-306. https://doi.org/10.1007/978-3-642-25664-6_35

[17] Atlam, H.F., Azad, M.A., Alassafi, M.O., Alshdadi, A.A. and Alenezi, A. (2020) Risk-Based Access Control Model: A Systematic Literature Review. *Future Internet*, **12**, 103. https://doi.org/10.3390/fi12060103

[18] Ueunten, K., Lum, C., Creigh, A. and Tsujita, K. (2015) Conservative Algorithms for Automated Collision Awareness for Multiple Unmanned Aerial Systems. *IEEE Aerospace Conference*, Big Sky, 7-14 March 2015, 1-18. https://doi.org/10.1109/AERO.2015.7118970

[19] da Silva, M.F., Honório, L.M., Marcato, A.L.M., Vidal, V.F. and Santos, M.F. (2020) Unmanned Aerial Vehicle for Transmission Line Inspection Using an Extended Kalman Filter with Colored Electromagnetic Interference. *ISA Transactions*, **100**, 322-333. https://doi.org/10.1016/j.isatra.2019.11.007

[20] Petráček, P., Walter, V., Báča, T. and Saska, M. (2020) Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization. *Bioinspiration & Biomimetics*, **16**, Article ID: 026009.

https://doi.org/10.1088/1748-3190/abc6b3

[21] Fulton, N.L., Baumeister, R., Westcott, M. and Estkowski, R.I. (2011) An Automated General Aviation Protection System for Manned and Unmanned Aircraft. *IEEE/AIAA* 30*th Digital Avionics Systems Conference*, Seattle, 16-20 October 2011, 5B2-1-5B2-16. https://doi.org/10.1109/DASC.2011.6096076

[22] McAree, O. and Chen, W.-H. (2013) Artificial Situation Awareness for Increased Autonomy of Unmanned Aerial Systems in the Terminal Area. *Journal of Intelligent & Robotic Systems*, **70**, 545-555. https://doi.org/10.1007/s10846-012-9738-x

[23] Radmanesh, M. and Kumar, M. (2016) Flight Formation of UAVs in Presence of Moving Obstacles Using Fast-Dynamic Mixed Integer Linear Programming. *Aerospace Science and Technology*, **50**, 149-160. https://doi.org/10.1016/j.ast.2015.12.021

[24] Di, B., Zhou, R. and Duan, H.B. (2015) Potential Field Based Receding Horizon Motion Planning for Centrality-Aware Multiple UAV Cooperative Surveillance. *Aerospace Science and Technology*, **46**, 386-397. https://doi.org/10.1016/j.ast.2015.08.006

[25] da Silva Arantes, J., da Silva Arantes, M., Fabiano Motta Toledo, C., Trindade Jr., O. and Williams, B.C. (2017) Heuristic and Genetic Algorithm Approaches for UAV Path Planning under Critical Situation. *International Journal on Artificial Intelligence Tools*, **26**, Article ID: 1760008. https://doi.org/10.1142/S0218213017600089

[26] da Silva Arantes, J., da Silva Arantes, M., Fabiano Motta Toledo, C., Trindade Jr., O. and Williams, B.C. (2017) An Embedded System Architecture Based on Genetic Algorithms for Mission and Safety Planning with UAV. *Proceedings of the Genetic and Evolutionary Computation Conference*, Berlin, July 2017, 10491056. https://doi.org/10.1145/3071178.3071302

[27] Leveson, N. (2013) An STPA Primer. MIT, Cambridge.

[28] Stringfellow, M.V., Leveson, N.G. and Owens Brandon, D. (2010) Safety-Driven Design for Software-Intensive Aerospace and Automotive Systems. *Proceedings of the IEEE*, **98**, 515-525. https://doi.org/10.1109/JPROC.2009.2039551

[29] Pappot, M. and de Boer, R.J. (2015) The Integration of Drones in Today's Society. *Procedia Engineering*, **128**, 54-63. https://doi.org/10.1016/j.proeng.2015.11.504

[30] Chen, J.Y., Zhang, S., Lu, Y. and Tang, P. (2015) STPA-Based Hazard Analysis of a Complex UAV System in Take-Off. 2015 *International Conference on Transportation Information and Safety*, Wuhan, 25-28 June 2015, 774-779. https://doi.org/10.1109/ICTIS.2015.7232133

[31] Lu, Y., Zhang, S.-G., Tang, P. and Gong, L. (2015) STAMP-Based Safety Control Approach for Flight Testing of a Low-Cost Unmanned Subscale Blended-Wing-Body Demonstrator. *Safety Science*, **74**, 102-113. https://doi.org/10.1016/j.ssci.2014.12.005

[32] Lum, C. and Waggoner, B. (2011) A Risk Based Paradigm and Model for Unmanned Aerial Systems in the National Airspace. *Infotech@Aerospace* 2011, St Louis, 29-31 March 2011, 1424. https://doi.org/10.2514/6.2011-1424

[33] Dempster, A.P. (1968) A Generalization of Bayesian Inference. *Journal of the Royal Statistical Society*: *Series B* (*Methodological*), **30**, 205-247. https://doi.org/10.1111/j.2517-6161.1968.tb00722.x

[34] Shafer, G. (1976) A Mathematical Theory of Evidence. Princeton University Press, Princeton.

[35] Murtha, J.F. (2009) Evidence Theory and Fault Tree Analysis to Cost-Effectively

Improve Reliability in Small UAV Design.

[36] da Silva Arantes, M., Toledo, C.F.M., Williams, B.C. and Ono, M. (2019) Collision-Free Encoding for Chance-Constrained Nonconvex Path Planning. *IEEE Transactions on Robotics*, **35**, 433-448. https://doi.org/10.1109/TRO.2018.2878996

[37] Yao, P., Wang, H.L. and Su, Z.K. (2015) Real-Time Path Planning of Unmanned Aerial Vehicle for Target Tracking and Obstacle Avoidance in Complex Dynamic Environment. *Aerospace Science and Technology*, **47**, 269-279. https://doi.org/10.1016/j.ast.2015.09.037

[38] Zhang, Z., Wu, J., Dai, J. and He, C. (2020) A Novel Real-Time Penetration Path Planning Algorithm for Stealth UAV in 3D Complex Dynamic Environment. *IEEE Access*, **8**, 122757-122771. https://doi.org/10.1109/ACCESS.2020.3007496