An Architecture to Automate UAS Operations in Non-segregated Airspace

Enric Pastor, Pablo Royo, Eduard Santamaria, Marc P. Batlle, Cristina Barrado, Xavier Prats
ICARUS Research Group
Technical University of Catalonia, Barcelona (Spain)

ABSTRACT

Technology evolution in the field of Unmanned Aircraft Systems (UAS) will affect the Air Traffic Management (ATM) performance regarding to new military and civil applications. UAS, as new airspace users, will represent new challenges and opportunities to design the ATM system of the future. The goal of this future ATM network is to keep intact (or improve) the network in terms of security, safety, capacity and efficiency level. Most UAS are, at present, designed for military purposes and very few civil applications have been developed mainly because the lack of a regulation basis concerning their certification, airworthiness and operations. UAS operations have always been solutions highly dependent on the mission to be accomplished and on the scenario of flight. The generalized development of UAS applications is still limited by the absence of systems that support the development of the actual operations. Moreover, the systematic development of UAS missions leads to many other operational risks that need to be addressed. All this elements may delay, increase the risk and cost in the implementation of a new UAS application.

Categories and Subject Descriptors

J.2 [**Physical Sciences and Engineering**]: *Aerospace.*

General Terms

Performance, Reliability.

Keywords

UAS automation, SOA architectures, Airspace integration.

1. INTRODUCTION

There is great pressure in order to define the rules under which UAS will be able to fly inside non-segregated airspace. This initial effort has been already started, mainly due to military interest [1]. A similar process will eventually happen for civil UAS, thus leading to the real introduction of UAS as an available product for science, business, etc.

EUROCONTROL and the FAA have similar philosophy about the integration problem: UAS should operate transparently to ATM and other airspace systems and users. However, a number of open issues must be addressed in order to obtain a successful UAS integration. Such situation will be extremely aggravated when UAS operational rules are introduced for the civil operation of UAS.

At present, the majority of manned flights correspond to commercial aviation dealing with persons/goods point to point transportation. On the contrary, the majority of potential UAS flight types may significantly differ from common manned flight types. Most common UAS potential mission is surveillance duties, requiring flexible and uncertain flight plans directly executed by computers with some supervision from UAS pilot. It is true that nowadays there are several general aviation manned aircraft performing this kind of missions, but its operation is mainly a man-directed process with little direct control from computers.

The introduction of this new type of unmanned traffic should not greatly affect ATM operations. However, UAS operation will be affected to large extends by its interaction with ATCs. Modern autopilots support pilots with re-planning capabilities, but only for point to point operations. Mission re-planning of surveillance UAS due to the integration in the non-segregated ATM systems will require lots of automated support for the UAS Pilot in Command (PiC) if a timely response by him is required.

It is also true that we can imagine in the future scheduled cargo or even eventually passenger UAS flights. This means that UAS integration in civil airspace will balance in some way the "general aviation flight types" with the "commercial aviation flight types" affecting to ATM operations and involved systems. However, the real integration of such type of flight will not occur in the short term, and therefore its study can be delayed until further UAS operational experience is gained.

Nowadays, no assessment exist dealing with the necessity to coordinate UAS almost automatic operations, but monitored by human pilots, with automatic or human operations performed by other airspace users and by the different ATM actors. Moreover, with the advent of civil UAS, the degree of automation will significantly increase because civil users won't be able to invest in extremely complex ground stations requiring multiple operators. Future integration of civil UAS should take into account relatively low cost but high automated vehicles.

Industry is currently designing and implementing the first family of *sense-and-avoid* systems [2,5]. Legally speaking these systems will allow the rightful operation

ATACCS'2011, 26-27 May 2011, Barcelona, Spain.

of UAS in non-segregated airspace. However, the separation provision and collision avoidance is hierarchically divided from the ATC to the pilot-in-command to the UAS autonomous operation. Therefore it is true that sense and avoid is a technical topic that must be successfully resolved, but it is also true that the UAS - ATM - Manned Aircraft triple interaction must be also addressed from a technological point of view, but also from an operational point of view.

This paper outlines an architecture designed to facilitate the automated operation of UAS, providing structural support to the PiC in order to implement both its intended mission, but also the integration of the UAS in non-segregated airspace. The architecture provides support for performing complex flight-plans, linking the UAS behavior to mission objectives, manage airfield operations and react to in-flight contingencies. Additionally, integration issues are supported by providing a coherent set of tactical and strategic reaction schemes (currently under development).

2. AERIAL WORK ORIENTED UAS

UAS have a great potential to support a wide variety of aerial monitoring applications. UAS may substitute manned aerial resources for cost/availability reasons; may cohabit with manned aerial resources in order to complement them; and even may allow addressing new monitoring scenarios in which manned platforms have never been introduced due to accessibility, complexity or risk. All these potential may be lost if all inherent risks in the UAS technology are not properly identified and addressed (see Figure 1).

The goal of UAS is to substitute manned aircraft in a number of aerial work scenarios. This is the first fundamental issue to take into account; UAS will not operate as point to point aircraft. Instead, UAS will possibly loiter over certain areas that may change over time. The main objective of the UAS Pilot in Command (PiC) being to attend to the commercial, security or scientific mission that the UAS is developing.

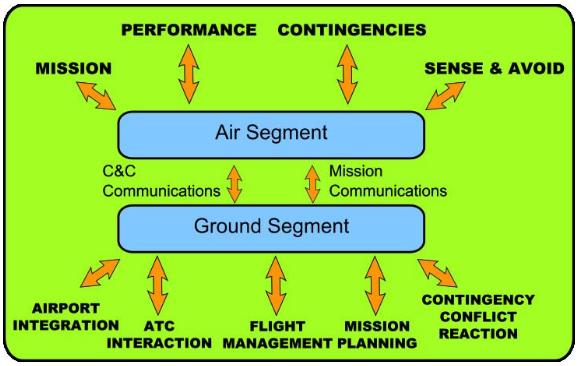

Figure 1. Overview of UAS Operational Open Issues.

Any change on the desired mission-oriented flight plan due to external interferences (ATC, traffic, etc.) will require the UAS PiC to redesign its operation to retake the tasks at hand prior to the undesired interruption.

Therefore *mission support* is required at the UAS in order to automate the operation, but also on the ground so that the PiC or the operator could manage the operation.

The operation of the UAS goes beyond basic point to point navigation. The UAS pilot will need to manage the trajectories that the vehicle will need to follow. This *flight management* may include the selection of alternative trajectories to implement departure and approach operations, or the selection of specific routes to respond to an optimum route selection.

Contingency reaction is also one of the main bottlenecks that will need to be addressed. In case of any type of contingency, from the vehicle or due to a conflict, an immediate reaction is mandatory in order to don't miss any precious second. Due to the limited situational awareness of the PiC, we advocate for pre-planned contingency reaction schemes associated to the flight plan itself.

Pre-planning for contingencies offers two main advantages: simplifies pilot decisions avoiding wrong selections due to the pressure of the circumstances, but also permits an automated contingency response in case the communication link between the ground and air segments is lost.

The desired goal by the UAS community is to allow them to operate in non-segregated airspace. UAS will need to *interact with the ATC* and with other aircrafts if operating in VFR airspace. Which and how are the flight intentions that UAS should provide to ATM actors? How and when these intentions will remain valid for the UAS and how they will have to be re-planned in flight in order to accommodate variations on the mission goals or to cope with variations induced by external events? Human factors are also considered crucial here. How the PiC will interact with the systems in order to react to these external events and how mission re-planning will be supervised by them?

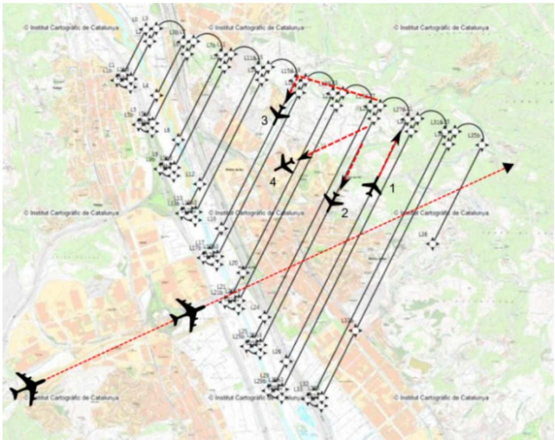

Figure 2. Separation and its impact over UAS missions.

Figure 2 shows a paradigmatic scenario. UAS will mostly perform scanning operations that, in case of security missions (due to disasters, fire, etc) may not be prevented even if other traffic operates through the area.. In this example, a UAS may be flying away from another flight (1), but all the sudden turns around and induces an unexpected conflict due to the scanning pattern (2). Instead of being diverted to some undesired location (4) it will be cleverer just to suggest ATC to skip a number of scan lines (3), rather than just cancelling the whole operation.

3. UAS MISSION ORIENTED ARCHITECTURE

The UAS System Abstraction Layer (USAL) is the set of available services running on top of the UAS system architecture to give support to most types of remote sensing UAS missions [3,4].

A number of specific services have been identified as "a must" in any real life application of UAS. The idea is to provide an abstraction layer that allows the mission developer to reuse these components and that provides guiding directives on how the services should interchange avionics information with each other.

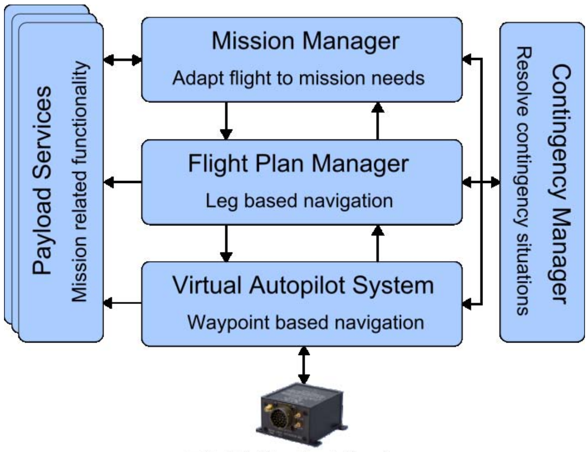

Figure 3. Overview of the USAL architecture.

Figure 3 conceptually describes the proposed separation between UAS mission, flight plan, and the underlying autopilot itself. Payload will be always commanded and exploited by the mission systems, while flight plan and/or telemetry information may be used by payload itself to localize information. The USAL services are divided in four categories [6]:

- Flight Services are those in charge of the UAS flight operation. This includes the autopilot management, flight management, flight monitoring for the PiC and the flight contingency management.
- Mission Services are those in charge of developing the actual UAS mission, controlling the payload and the area of surveillance, processing or saving data and showing it to PiC.
- Awareness Services are in charge of the safe operation of the UAS with respect terrain avoidance and integration with shared airspace.
- Payload Services are lower level services, not necessarily available to the end-users. They are like device-driver, this is, the facility services that abstract the details to access to the input, output and communication devices.

Flight Services

Flight services are a set of USAL applications designed to properly link the selected UAS autopilot with the rest of the UAS avionics [6,7]. The main services operated are the Virtual Autopilot Service, the Flight Manager Service, the Contingency Service, the Flight Monitor Service, the Flight Plan Monitor Service etc. (see Figure 4).

The Virtual Autopilot Service (VAS) is a system that interacts with the selected autopilot and is adapted to its peculiarities. The VAS abstracts the implementation details from actual autopilot users. From the mission/payload subsystems point of view, the VAS is a service provider that offers a number of standardized information flows independent of the actual autopilot being used.

The Flight Plan Manager (FPMa) is a service designed to implement much richer flight-plan capabilities on top of the available autopilot capabilities. The FPMa offers a virtually unlimited number of waypoints, waypoint grouping, structured flight-plan phases with built-in emergency alternatives, mission oriented legs with a high semantic level like repetitions, parameterized scans, etc.

These legs can be modified by other services in the USAL by changing the configuration parameters without having to redesign the actual flight-plan; thus allowing the easy cooperation between the autopilot and the UAS mission.

The Contingency Management services [8] monitor critical parameters of the operation (like battery live, fuel, flight time, system status, etc.). In case contingencies are detected, actions will be taken in order to preserve the security and integrity of the UAS: from flight termination, mission abort or system re-cycle.

The Electrical and Engine Management services are a set of services designed to gather data on the operation of the UAS electrical system and the propulsion system. Such information is relayed to the Contingency Manager to take the appropriate decisions.

The Flight Termination System is a system outside the USAL architecture, and it is in charge to deploy a parachute system in case the Contingency Manager requires it; also the parachute may be deployed in case a major USAL failure.

The Flight Plan Monitor is the HMI interface on the ground that provides high level flight management services that will exploit the advanced capabilities offered by the UAS oriented flight plan provided within USAL.

Awareness Services

A UAS is a highly instrumented aircraft with no pilot on board. The most suitable flight rules for a UAS are IFR (Instrumental Flight Rules), however for remote sensing missions the advantages of UAS systems is precisely its capacity for flying at any altitude, where VFR (Visual Flight Rules) aircrafts are found.

UAS must rely on its equipment to properly inform the PiC, or substitute the pilot capacities in VFR conditions. Flight services are in charge of the aircraft management in normal conditions, while the Awareness Services are in charge of monitoring surroundings conditions and overtake aircraft management in critical conditions (see Figure 4). In this case mission services come to a second priority, until flight conditions become again normal.

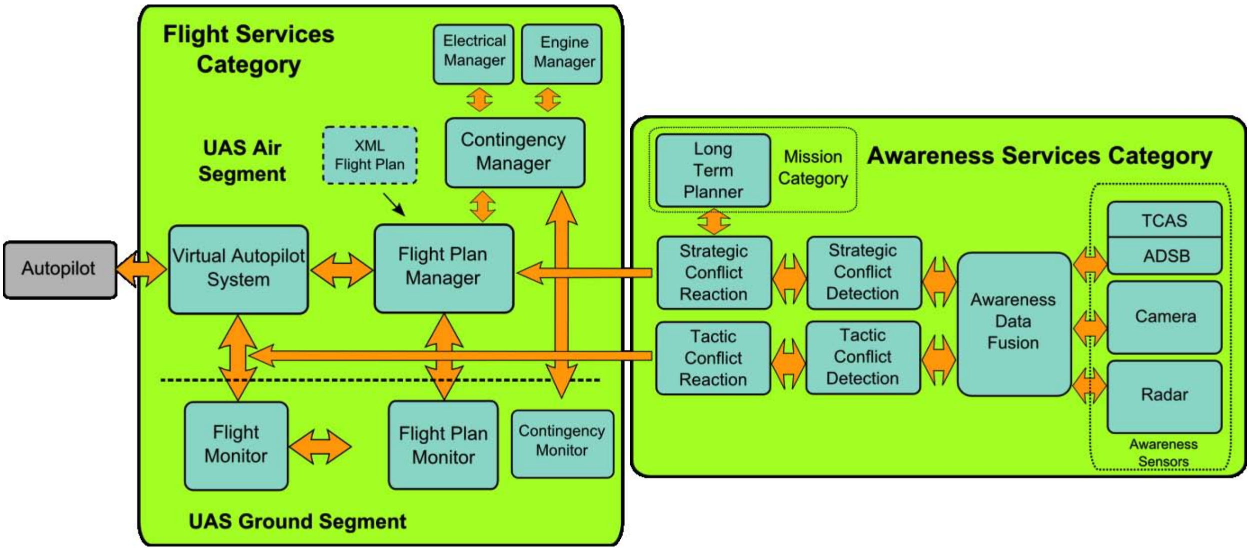

Figure 4. Overview of the Flight and Awareness Services.

The Awareness data fusion (ADF) is a service designed to collect all available data about air vehicles surrounding our UAS, terrain and meteorological conditions. All this information can be obtained either by on board sensors or even through an external provider.

The Tactical/Strategic Conflict Detection service will analyze the fused information offered by the ADF in order to detect potential collision conflicts with objects/terrain/bad climate. Depending on the type of conflict, different types of reaction procedures will be activated. While reaction is executed it will keep monitoring than the conflict is really being avoided.

The Tactical/Strategic Reaction services, will implement avoidance procedures according to the severity of the conflict. Tactical reaction is designed in such a way it can overtake the Flight Plan Manager in order to execute a radical avoidance maneuver. Once completed, the FPMa will regain control. A strategic reaction will command the FPMa to slightly modify its selected flight plan trying to avoid the conflict but at the same time retaining the original mission requested by the Mission Manager.

4. UAS FLIGHT PLAN SPECIFICATION

The flight plan is a document that contains the navigation instructions for the UAS [7,9]. In our proposal the flight plan is a self-contained description of the main flight plan, but also contains options for take-off and landing operations as well as alternatives for emergency situations (see Figure 5).

Stages constitute high-level building blocks for flight plan specification and are used to group together legs that seek a common purpose. They correspond to flight phases that will be sequentially executed: Taxi, TakeOff, Departure, EnRoute, Mission, Arrival, Approach and Land.

A *leg* specifies the flight path to get to a given waypoint. Legs contain a destination waypoint and a reference to their next. Most times legs will be flown in a single direction, but within iterative legs reverse traversal is also supported.

There are four kinds of legs. Basic legs to specify basic traditional primitives; Iterative legs to specify repetitive sequences; Intersection legs that provide a junction point for legs which end at the same waypoint, or a forking point where a decision on what leg to fly next can be made; and parametric legs that specify legs whose trajectory can be computed given the parameters of a generating algorithm, e.g. a scanning pattern.

A complex trajectory may involve iteration, thus the inclusion of iterative legs. An iterative leg has a single entry (i.e. its body can be entered from a single leg), a single exit and includes a list with the legs that form its body. Every time the final leg is executed an iteration counter will be incremented. When a given count is reached or a specified condition no longer holds the leg will be abandoned proceeding to the next one.

Intersection legs are used in situations where there is more than one possible path to follow and a decision needs to be made. This leg type contains a list with the different alternatives and a condition for picking one of them. Intersection legs are also used to explicitly indicate where two or more different paths meet.

Together with parametric and iterative legs, intersection legs provide a powerful means for adapting the flight as best suited to the ongoing mission circumstances.

With parametric legs complex trajectories can be automatically generated from a reduced number of input parameters. If the actual values of these parameters change, the resulting trajectory will be dynamically recomputed. Eventually a complete enough library of different parametric legs will be available so that a wide range of missions can be performed.

Analysis of the potential contingency situations and planning the correct reaction is a critical task to be carried out by any airplane to guarantee its safe operation. Pilot's reactions to any kind of incidences that may occur in-flight are critical, and will determine the fate of the flight in case such contingency occurs.

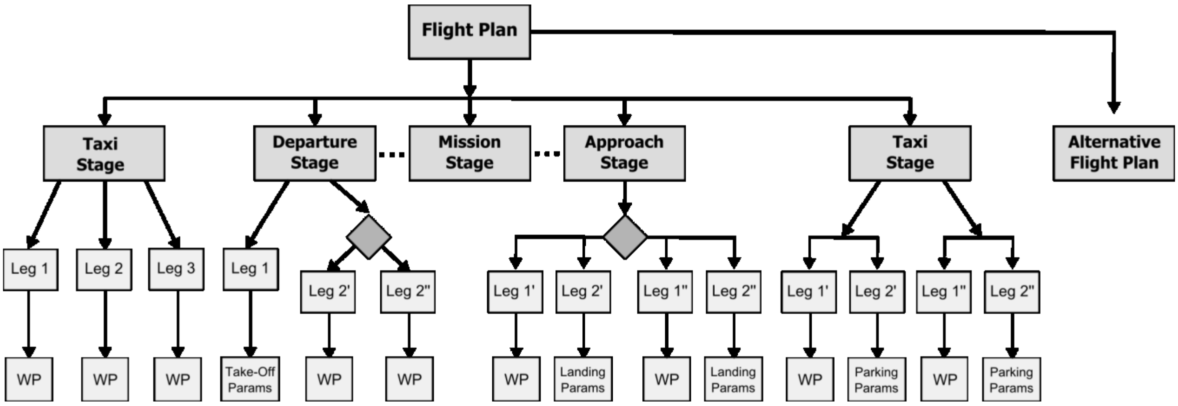

Figure 5. Structure of a USAL flight plan with embedded airfield and alternative parameters.

5. IN FLIGHT CONTINGENCY MANAGEMENT

Contingency management relates to the capability of the system to monitor its health status, detect anomalies and react accordingly. During a pre-flight phase all reasons that may lead to a deviation from the expected UAS behavior are identified and assigned a pre-defined reaction. USAL introduces a contingency architecture [8], that is built by two components: the Health Monitor (HM) and the Contingency Intelligent Control (CIC).

The HM gathers and processes the information needed to take a contingency decision. The CIC is in charge of deciding the proper response or set of responses for dealing with a particular contingency. The CIC classifies the contingency into three categories: Minor, Hazardous and Catastrophic (see Figure 6).

Catastrophic Contingencies includes all contingencies which interrupt the UAS flight or a safety landing. In practice it means loss of the platform.

In order to respond to these contingencies, it is considered an emergency component aggregated to our architecture called *Flight Termination System* (FTS).

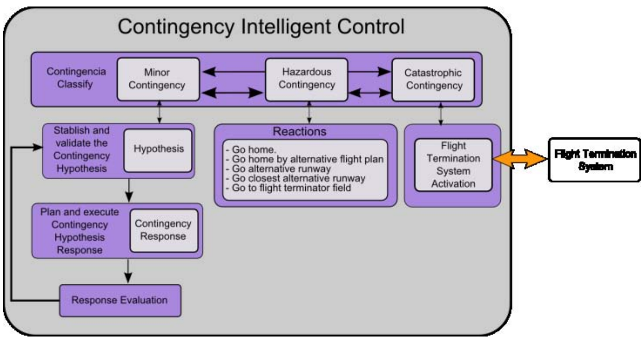

Figure 6. Architecture of the contingency reaction system.

The FTS commonly will be composed by parachute system [10]. The main objective is to guarantee that the potential impact to the ground of the UAS will not fatally damage any person or infrastructure.

Hazardous Contingencies includes all contingencies which reduce the aircraft airworthiness. This lack of airworthiness may put in danger the mission success or sometimes develop into catastrophic contingency. Also this category is composed by those contingencies which make impossible the mission objectives, as for example any failure in the payload needed for the mission. This component has different reactions in front of these contingencies.

- *Go Home*: The UAS will be sent directly to its final destination and the mission will be aborted.
- *Go Home by Alternative Flight Plan*: The UAS will flight back home. If the emergency situation in critical enough, it may be needed an alternative path which description is composed by alternative paths; these paths are managed by the Flight Plan Manager.
- *Go Better Alternative Runway*: A UAS flight plan presents different landing possibilities. Due to its little size a lot of airfields may be suitable enough to ensure safety landings. This response is focused in finding the best alternative runway.
- *Go Closest Alternative Runway*: A landing site is needed as soon as possible in order to preserve the UAS platform.
- *Go to Flight Termination Field*: If the UAS cannot arrive to the closest runway, it must find somewhere to terminate the flight.

6. MISSION AND PAYLOAD SERVICES

The goal of the Mission Management is to extend the UAS automation capabilities by being able to execute a specification of the UAS behavior through a work-flow like mechanism. This specification determines how operation of on-board services is orchestrated during a mission. USAL implements a reconfigurable MMa service based on the Harel's Statecharts [11] formalism for concurrent systems. The overall Mission and Payload service architecture is outlined in Figure 7.

The use of state machines for specifying behavior is not new, examples can be found in [12] and [13]. The MMa implements a multi-faceted mechanism that supports coordinating mission objectives, PiC requirements, actual flight of the UAS, payload operation and supporting mission services.

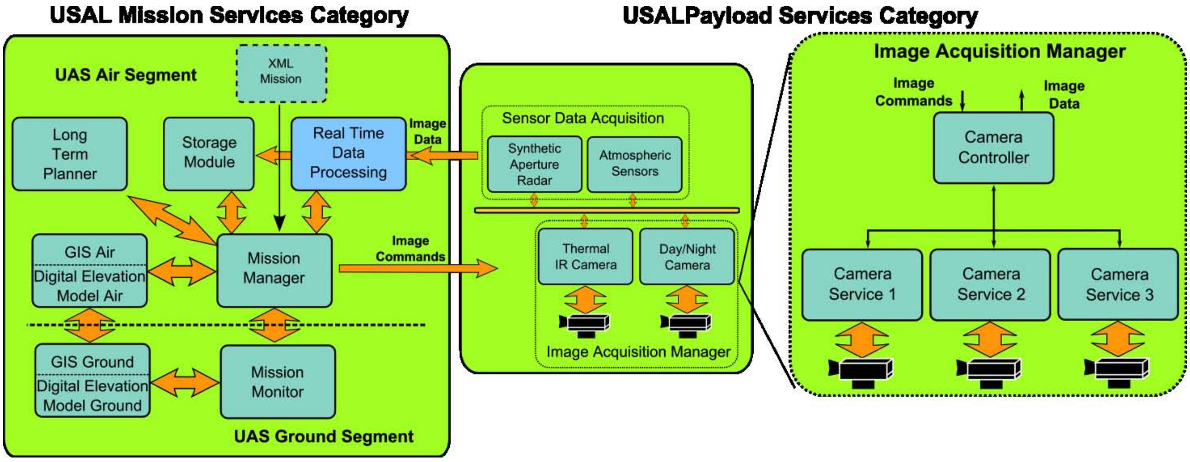

Figure 7. Overview of the Mission and Payload Services.

Statecharts and architecture of the MMa service

Statecharts can be used to model the behavior of complex reactive systems. Graphically, a state diagram is a collection of nodes, representing states, and arcs, representing transitions. Common characteristics of Statecharts are the following:

- A state reflects the current configuration of the system.
- A transition is a relationship between two states. It indicates that a system in the first state will enter the second state when a specified event occurs and the specified guard conditions are satisfied.
- The events that cause a reaction are called triggers.
- A condition used to specify under what circumstances a given transition is permitted.
- Transitions can be accompanied by actions to be performed during the transition. Typical things actions are used for include firing another event, updating some data structure and interact with the outside world.

The language and underlying execution environment used to describe the desired mission work-flow are State Chart XML (SCXML) [14]. SCXML is a working draft that provides a general purpose event-driven state machine language based on Harel's Statecharts [20] as part of the Unified Modeling Language (UML) [15].

Missions are specified in USAL by combining two elements: an SCXML diagram specification, and a number of user defined software functions that are associated to every state and transition. These functions are executed once a particular state is reached or a transition executed. Figure 8 depicts a high level view of this organization.

The MMa service is organized around an SCXML execution engine that is wrapped up by a number of additional components and a middleware. The SCXML engine allows the interpretation of the mission automaton according to the surrounding events. For each state or transition, user-defined software is executed. This code will include all the necessary invocations to supporting USAL services through the standardized middleware. External invocations may include the desired real-time flight plan updates in order to satisfy the actual mission requirements; but also triggering all the necessary data gathering (by turning on payload services); or transforming data into high level information through external real time computation services. Additionally, if highly precise trajectories are required, external trajectory planning services can also be invoked.

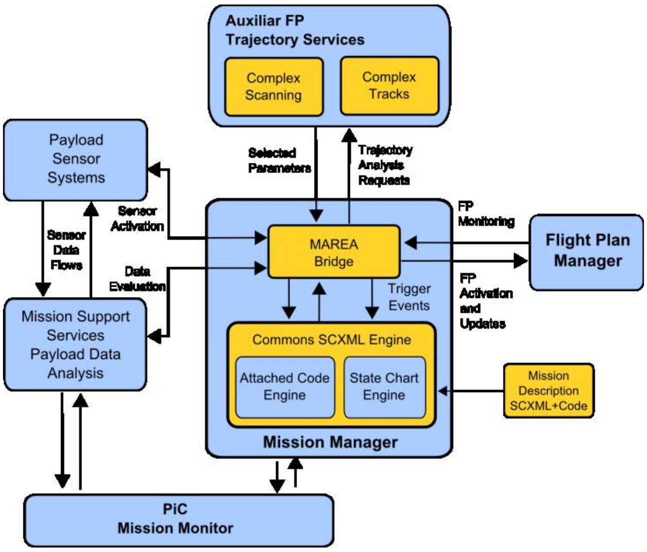

Figure 8. Architecture of the MMa service within USAL.

The proposed strategy offers a number of pre-defined messages that allow a built-in coordination between the MMa and the FPMa. On one side, the FPMa keeps the MMa informed about which specific Stage or Leg is being flown. These messages translate into events that feed the MMa execution engine. As a result of these incoming events, the MMa may change its current state generating some kind of response in the process. A response may consist in a message modifying the behavior of some UAS service (mission or payload related), or even the flight plan.

7. SIMULATED EVALUATION USE CASE

In this section, we provide the results of a simulated hot-spot detection mission that is backed-up by a real helicopter-based UAS designed to implement the same operation. The goal of the mission is to inspect a burned area just after fire containment and detect remaining hot-spots which could potentially revive the fire front. Automation of this type of mission would permit minimizing the amount of valuable firefighting resources required for this task [16].

The mission consists in scanning the burned area following a classical scan pattern, although more elaborated perimeter scanning schemes can be employed by using the appropriate parametric leg. During the scan, thermal imagery is taken that is then processed on board the UAS to detect potential hot-spots. Each potential hot-spot should be further analyzed by flying an eight pattern over it (or by holding if flown by a helicopter) to determine whether it represents a real threat.

To demonstrate the flexibility provided by the proposed architecture and mission management approach, the hot-spot detection mission is performed employing two different strategies, both of them exploiting the same flight plan template in different ways.

In the first case, it is assumed that a significant computation time is required to process the recorded thermal imagery. Therefore, there is a long delay between the point when a thermal image is taken and the moment when it is determined that it contains a potential hot-spot (and its exact location computed). To avoid unnecessary delays, a full scan is executed first collecting thermal imagery. During the scan process thermal information is analyzed and if hot-spots detected queued to be visited later. Once the scanning process is completed each potential hot-spot detected so far is visited in optimal sequence. Hot-spots can be added in the visiting sequence until all acquired imagery has been fully analyzed, so the visiting sequence may change every time a new hot-spot is detected in this phase.

In the second case, we assume that more capable computation services are available and therefore almost real-time hotspot detection on-board the UAS is possible. We take advantage of this capability to fly an eight pattern intermediately upon hot-spot detection and then resume the scanning of the area after the detailed inspection of the potential hot-spot has been completed.

A potential dynamic selection of legs by the MMa is depicted in Figure 10. In this fragment it can be seen how the MMa will select a specific mission behavior by sending a stream of leg updates and leg selections according to the desired strategy and detected hot-spots. In this way, even though the actual UAS behavior may be quite complex and elaborated, simplicity is maintained both in the flight-plan and mission descriptions, but the interleaving of both specifications results into a rich and powerful mechanism.

Underlying flight plan template

The underlying flight plan is common to both versions of the hot-spot mission (see Figure 9). The UAS can either perform a scan (scanArea leg), an eight pattern (scanPoint leg), a holding pattern (hold leg) or a short navigation from one area of interest to another (navigation leg).

Which leg is actually selected depends on the condition of an intersection leg called patternSelect. The four alternatives converge at another intersection leg called join. Finally an iterative leg called loop is used to enable the UAS to alternate between the different options.

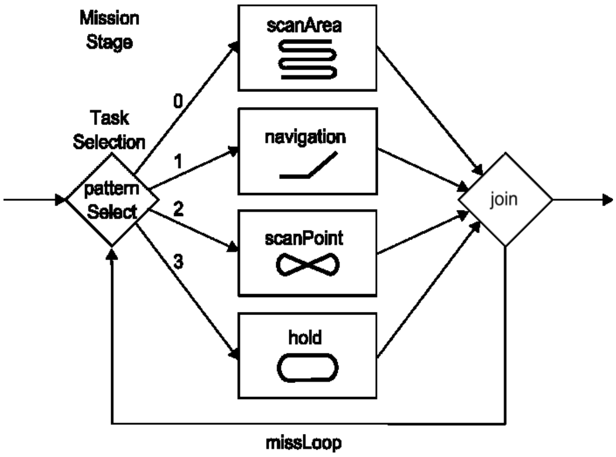

Figure 9. Mission fragment of a flight plan.

Specification of the UAS' mission behavior

In the so-called deferred hot-spot analysis version of our mission, the expected behavior that the PiC will command a full scan of the area of interest at the beginning of the operation. During the flight thermal images are acquired and processed, hot-spots detected and analyzed later on in a second phase. Figure 11 shows the statechart that refines the Mission state.

When the Mission state is reached an initial configuration state is entered (ConfPayload) in which payload and additional services are started-up. An external supporting service devoted to image processing (for hot spot detection) will be subscribed to the thermal images provided by the UAS payload. This data flow will be set up during state ConfPayload and activated when required.

After initial configuration, two parallel sub-states are simultaneously entered: HotSpotsCounter, used to keep track of the encountered potential hot-spots, and ScanArea, to support the systematic sweep the area of interest.

The operation within all MMa states that have a direct relation with a FPMa legs is designed in a similar way:

1. Upon entering a state, we set the result of the selection condition to control which leg will be flown next.
2. Then, if some event is received that requires the initial decision to be reconsidered, we use the triggered transition to update the flight plan and change the selection. Updates can be even performed multiple times changing the UAS planned operation taking into consideration the full set of events up to that point.

The HotSpotsCounter state

Following this principle, the operation of the HotSpotsCounter state is as follows. Each time a hotspot event is delivered, the HotSpotsCounter state will loop

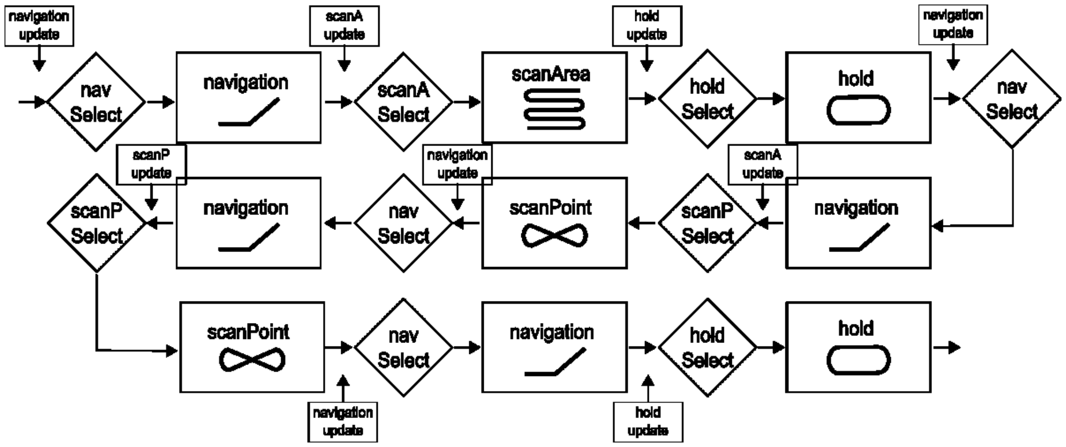

Figure 10. Leg execution resulting from the selected mission work-flow.

over itself, executing code that will collect the hot-spot information and queue it following a priority scheme or an optimum exploration scheme. A dedicated event will be received once all thermal images have been fully processed, which will signal the MMa that the hot-spot exploration has been completed. When the hotspot event happens, at least one potential hot-spot needs to be visited. During the HotSpotsCounter's transition we also re-schedule the optimum order to visit the detected hot-spots. Also, we update the coordinates of the scanPoint leg to the first non-visited potential hot-spot and modify the selection condition in patternSelect so that scanPoint is picked.

The ScanArea state

Within the ScanArea a potential hot-spot is detected, and the flight plan is updated to perform a scanPoint operation over the first pending hot-spot. If no hot-spot was detected, the flight plan will be configured to perform a Hold.

Once the scanPoint or hold leg starts its execution, the FPMa will notify the MMa through the standardized events defined by the USAL. This notification will, in turn, trigger a transition from the ScanArea state to the ScanPoint state or the hold state.

The ScanPoint state

The operation of the ScanPoint state is as follows. When entering the state, if no more hot-spots remain to be visited the flight plan will be updated so that a hold operation will follow. If remaining hot-spots exists the flight plan will be updated to analyze the next one. If a new hot-spot is detected and notified through the HotSpotsCounter state. In that case, from that state, the flight plan will be updated again to take into account the new detected hot-spot. In the end we will observe that the ScanPoint state may transition again to itself or to the Hold state.

The Hold state

The MMa will enter into the Hold without any pre-defined assumption. If within this state it is identified that no more hot-spots should be analyzed, the state will simply transition into itself, updating the location in which the holding operation is performed to be ready to execute another scanning operation. If no further scanning is required, the system can abandon the mission area in order to follow the returning route.

If while holding, another hot-spot is detected, the flight plan will be updated through the HotSpotsCounter state in order to transition to the ScanPoint state. Note that, as the hold leg never really ends, a skip message will be sent to the FPMa to force the leg change.

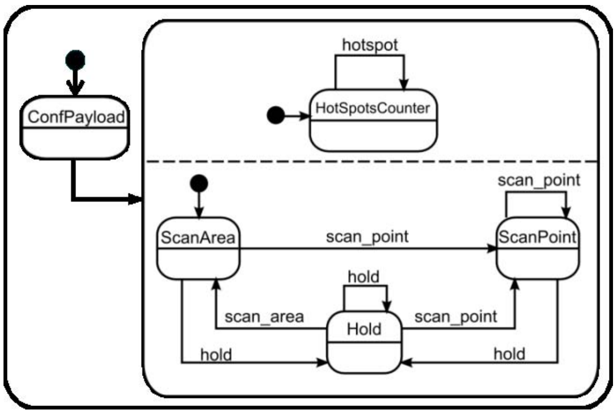

Figure 11. Mission state for deferred hot-spot analysis.

8. EXPERIMENTAL EVALUATION

To carry out the execution of the mission a simulation environment has been set up. The aircraft behavior is simulated using the FlightGear Flight Simulator [17]. With the VAS service handling all interactions with FlightGear, the rest of the system is completely unaware of the fact that the flight is simulated. Both the USAL FPMo and a Google Earth client can be used for tracking the UAS flight and provides real time visualization of the mission evolution.

Figure 12 shows the trajectory of the aircraft when detailed analysis of potential hot-spots is deferred. Bonfire icons indicate the position where the potential hot-spots are located. The aircraft performs a full scan of the area of interest. Each time a hot-spot is detected the MMa is notified and this triggers a self-transition on the *HotSpotsCounter* state. During this transition the number of potential hot-spots detected is incremented. When the scan finishes, execution of the scanPoint leg on the first hot-spot starts, triggering a transition from ScanArea to the ScanPoint state. During this transition the number of visited

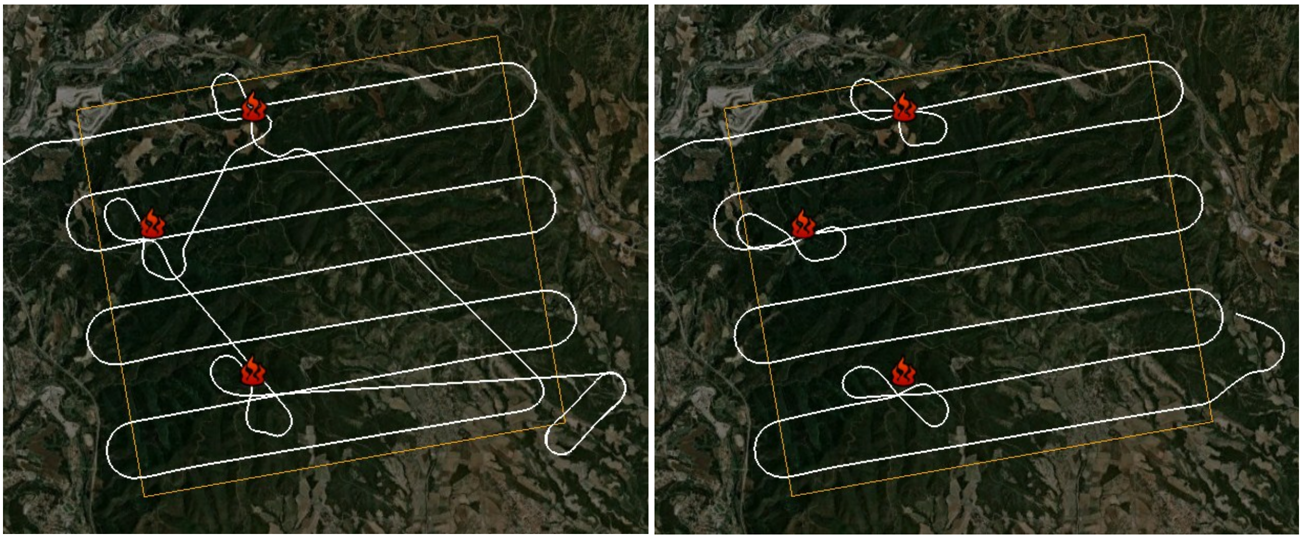

Figure 12. Aircraft trajectory with deferred and inmediate analysis.

hot-spots is incremented and the scanPoint leg is updated with the coordinates of the next hot-spot in the pending queue. The UAS executes as many scanPoint legs as detected potential hotspots. Figure 12 also shows the results of the immediate hot-spot detection. At each hot-spot an eight pattern is immediately flown.

This mission concept has been employed to implement a helicopter-based UAS called Sky-Eye (see Figure 13). This system is designed to improve the overall awareness of the fire managers by providing tactical support to wildfire monitoring and control of ground squads [16]. The Sky-Eye prototype is built around the AP04 autopilot and existing commercial off the- shelf (COTS) technology that can be immediately deployed on the field at a reasonable cost. Sky-Eye is designed to increase the level of UAS automation while being controlled from a mission point of view by a PiC. Information is gathered by the on-board cameras, processed and then relayed following the described strategies in such a way that it can be immediately exploited by the fire fighter squads.

Figure 13 shows the Sky-Eye aerial-segment prototype, including the AP04 autopilot, a number of embedded computers to properly manage the USAL architecture, a local area network to link computational nodes with sensors, several cameras, data storage, communications devices, image processing units and mission management capacities. The airframe is a VARIO RC helicopter enlarged with larger carbon fiber skids to hold the payload.

Sensors are basically the two cameras installed on board: a high definition visual camera, and a thermal camera. Additionally, there is a video camera placed in front of the fuselage to support take-off and landing operations. This platform has been selected according to the targeted Mediterranean wildfires and to the cost objectives. A low/medium altitude tactic UAS was preferred for cost availability and due to limited fire sizes. A WLAN communication (802.11a, 5GHz) is used for mission/payload communications while a 900MHz dedicated link is used to route command and control. USAL employs built-in routing mechanisms to direct the appropriate USAL messages to each specific link.

The Sky-Eye development has been greatly simplified thanks to USAL architecture. Initial prototypes were implemented by using the simulated version of the AP04. An almost immediate migration was possible to the real AP04, while a fixed wing aircraft version is currently under development using another commercial AP unit (and exploring a different set of monitoring strategies). Thanks to the USAL concept the overall mission oriented architecture will be migrated from a tactical to a strategic monitoring platform with little effort.

9. CONCLUSIONS

This paper has reviewed a number of issues that limit the integration of UAS in non-segregated airspace. These factors relate to the fact that UAS operate as mission-oriented vehicles rather than point to point transportation.

In order to address these factors, an UAS oriented architecture has been introduced. This architecture supports the development of mission-oriented flight-plans with embedded alternatives to manage departure and approach operations. The architecture also supports embedded contingencies so that the PiC can supervise semi-automatic reactions, or the UAS can automatically react as pre-planned in case the control link is lost.

Future work will address the analysis of the automatic reaction to both tactical and strategic aerial conflicts, and how the mission-oriented flight path can be retaken after conflicts are resolved.

ACKNOWLEDGMENTS

This work has been partially funded by Ministry of Science and Education of Spain under contract CICYT TIN 2007-63927. This work has been also co-financed by the European Organization for the Safety of Air Navigation (EUROCONTROL) under its CARE INO III programme. The content of the work does not necessarily reflect the official position of EUROCONTROL on the matter.

Figure 13. Sky-Eye UAS prototype implementing hot-spot missions based on the USAL.

REFERENCES

1. European Organisation for the Safety of Air Navigation. EUROCONTROL specifications for the use of military unmanned aerial vehicles as operational air traffic outside segregated airspace, 2006.
2. European Organisation for the Safety of Air Navigation. Unmanned Aircraft Systems ATM Collision Avoidance Requirements, 2010.
3. Earth Observations and the Role of UAVs: A Capabilities Assessment. Version 1.1, NASA's Civil UAV Assessment Team, 2006.
4. EU Civil UAV Roadmap, USICO, 2005.
5. DO-304: Guidance Material and Considerations for Unmanned Aircraft Systems, RTCA, 2007.
6. E. Pastor, C. Barrado, P. Royo, J. Lopez, E. Santamaria, An open architecture for the integration of UAV civil applications, in: T. M. Lam (Ed.), Aerial Vehicles, IN-TECH, 2009, pp. 511–536.
7. Santamaria E, Royo P, Lopez J, Barrado C, Pastor E, and Prats X. Increasing UAV capabilities through autopilot and flight plan abstraction. Proc Proceedings of the 26th Digital Avionics Systems Conference, Dallas, Texas, 2007. AIAA.
8. E. Pastor, P. Royo, E. Santamaria, X. Prats, C. Barrado, In-flight contingency management for unmanned aerial vehicles, in: Proceedings of the AIAA Unmanned...Unlimited Conference, AIAA, Seattle, Washington (USA), 2009.
9. X. Prats, L. Delgado, P. Royo, M. Perez-Batlle, E. Pastor, Depart and approach procedures for UAS in a VFR environment, Journal of Aircraft (In press).
10. Stansbury R, Wilson T, and Tanis W. A technology survey of emergency recovery and flight termination systems for uas. Proc Proceedings of the AIAA Infotech@Aerospace Conference and AIAA Unmanned...Unlimited Conference, Seattle, OR, 2009. AIAA.
11. D. Harel, M. Politi, Modeling Reactive Systems with Statecharts: The STATEMATE Approach, McGraw-Hill, 1998.
12. D. Mackenzie, R. Arkin, J. Cameron, Multiagent mission specification and execution, Autonomous Robots 4 (1997) 29–52.
13. J. Boren, S. Cousins, The smach high-level executive, IEEE Robotics & Automation Magazine 17 (2010) 18–20.
14. W3C Draft: State Chart XML (SCXML) State Machine Notation for Control Abstraction, World Wide Web Consortium (W3C), 2010. http://www.w3.org/TR/scxml/.
15. G. Booch, J. Rumbaugh, I. Jacobson, Unified Modeling Language User Guide, The (2nd Edition) (The Addison-Wesley Object Technology Series), Addison-Wesley Professional, 2005.
16. E. Pastor, C. Barrado, P. Royo, E. Santamaria, J. Lopez, E. Salami, Architecture for a helicopter-based unmanned aerial systems wildfire surveillance system, Geocarto International 26 (2011) 1–19.
17. FlightGear Flight Simulator, 2010. Http://www.flightgear.org.